Министерство науки и высшего образования Российской Федерации
ФГБОУ ВО «Удмуртский государственный университет»
Институт языка и литературы
Кафедра профессионального иностранного языка
для естественнонаучных специальностей

Cybernetics & Computer science
for Math students

Учебно-методическое пособие

Ижевск
2025

УДК 811.111’36 (075.8)
ББК 81.432.1-2я73
С98

Рекомендовано к изданию Учебно-методическим советом УдГУ

Рецензент: канд. пед. наук, доцент, зав. кафедрой ФГБОУ ВО «УдГУ» ЛиЛСИПК **Е.В. Тарабаева.**

Составитель: Касаткина Т.Ю.

С98 Cybernetics & Computer science for Math students : учеб.-метод. пособие / сост. Т.Ю. Касаткина. – Ижевск : Удмуртский университет, 2025. – 1 Мб. – Текст : электронный.

Пособие предназначено для студентов Института математики, информационных технологий и физики для изучения дисциплины «Иностранный язык в академической и профессиональной коммуникации» по направление подготовки высшего образования – магистратура 02.04.01.01 Математическая кибернетика. Материалы пособия ориентированы на формирование навыков и умений перевода профессиональных текстов с английского языка на русский, их реферирования и пересказа. Издание включает задания на расширение профессионального словарного запаса студентов, повторение грамматических явлений, которые встречаются в представленных для изучения текстах.

Пособие рекомендуется использовать на аудиторных занятиях по английскому языку, для самостоятельного изучения тем, а также для переводческой практики студентов, получающих дополнительную квалификацию по направлению «переводчик в сфере профессиональной коммуникации».

Минимальные системные требования:
Celeron 1600 Mhz; 128 Мб RAM; Windows XP/7/8 и выше, 8x DVD-ROM
разрешение экрана 1024×768 или выше; программа для просмотра pdf.

Касаткина Татьяна Юрьевна
Cybernetics & Computer science for Math students
Учебно-методическое пособие

Подписано к использованию 18.12.2025
Объем электронного издания 1 Мб
Издательский центр «Удмуртский университет»
426034, г. Ижевск, ул. Ломоносова, д. 4Б, каб. 021
Тел. : +7(3412)916-364 E-mail: editorial@udsu.ru

CONENTS

UNIT 1. CYBERNETICS ... 5
Text 1. Definition of cybernetics ... 5
Text 2. Historical development of cybernetics ... 5
Text 3. Cybernetics today ... 8
Text 4. Subdivisions of cybernetics. Part 1. ... 11
Text 4. Subdivisions of cybernetics. Part 2. ... 14
Text 5. The relation of cybernetics to mathematics ... 17
Text 6. Interrelation between cybernetics and modern computer technology ... 20
Text 7. Cybernetics as a science ... 21
UNIT 2. COMPUTER SCIENCE ... 24
Text 1. The development of computer science ... 24
Text 2. Major branches of computer science ... 28
Text 3. Architecture ... 32
Text 4. Artificial intelligence, robotics, human-computer interfacing ... 37
Text 5. Basic features of database programs ... 40
Text 6. Computer graphics ... 42
Text 7. Programming languages ... 43
Text 8. Debugging a computer program ... 48
Text 9. Software testing ... 50
Text 10. Computer crimes ... 54
Text 11. What is information security ... 57
Text 12. Machine translation today and tomorrow ... 61
UNIT 3. PECULIARITIES OF TRANSLAYING MATH TEXTS ... 64
Стилистические особенности математических текстов ... 64
Способы преодоления интерференции при переводе текстов по математике ... 65
Ложные друзья переводчика ... 70
Перевод имён собственных (Proper nouns) ... 72
Перевод наиболее употребительных словосочетаний с предлогами и союзами в математическом английском ... 73
Предлоги, союзы и обороты, характерные для научной речи ... 74
Методические указания студентам по чтению и переводу профессионально-ориентированных текстов ... 74
Список литературы ... 77

Предисловие

Учебно-методическое пособие по чтению и переводу профессиональных текстов предназначено для студентов-магистрантов Института математики, информационных технологий и физики для изучения дисциплины «Иностранный язык в академической и профессиональной коммуникации». Издание соответствует программным требованиям Федерального государственного образовательного стандарта для преподаваемой дисциплины.

Актуальность создания данного пособия обусловлена тем, что в нём представлен материал профессиональной направленности, который способствует формированию коммуникативной, когнитивной и лингвистических компетенций у студентов-магистрантов.

Темы, предложенные в разделах данной работы, помогут студентам ориентироваться в английской научной литературе, овладеть терминологией в области кибернетики и компьютерных наук, освоить новый языковой материал. Материал пособия способствует формированию таких универсальных компетенций, как умение анализировать аутентичный материал, извлекать информацию для дальнейшего практического применения, умение понимать узкоспециальную литературу, составлять словарь по IT терминам, участвовать в обсуждении тем, связанных со специальностью.

Каждый текст снабжён предтекстовыми словами и выражениями для самостоятельного перевода и дальнейшего заучивания. Профессионально-ориентированные тексты также направлены на совершенствование навыков различных видов чтения и расширение словарного запаса обучающихся.

После каждого текста представлены лексико-грамматические упражнения. Их количество зависит от сложности самого текста, лексики и грамматики. В пособии грамматический материал повторяется студентами при переводе текстов. При необходимости можно обратиться к теоретическому разделу пособия.

Учебно-методическое пособие «Cybernetics and Computer science for Math students» подходит для решения следующих учебных задач: во-первых, научить студентов читать профессиональную литературу, извлекая при этом научную информацию с нужной степенью полноты и точности, во-вторых, переводить аутентичные тексты с английского языка на русский язык, в-третьих, достичь определённого уровня владения устной речью, который позволил бы студентам вести беседу по специальности и делать устные научные сообщения.

Касаткина Татьяна Юрьевна,
кандидат филологических наук, доцент,
доцент кафедры ПИЯЕНС, ИЯЛ.

UNIT 1. CYBERNETICS

TEXT 1. Definition of cybernetics

Just like any comprehensive category, cybernetics would hardly possess a unique definition. Moreover, the meanings of terms describing this category also evolve with the course of time. Let us give a series of widespread definitions of cybernetics:

1. "A science concerned with the study of systems of any nature which are capable of receiving, storing, and processing information so as to use it for control"– ***A. Kolmogorov;***
2. "Scientific treatment of the system in which complexity is outstanding and too important to be ignored."– ***W. Ashby;***
3. "A branch of mathematics dealing with problems of control, recursiveness, and information, focuses on forms and the patterns that connect."– ***G. Bateson;***
4. "The art of effective organization."– ***S. Beer;***
5. "The art of securing efficient operation."– ***L. Couffignal;***
6. "The art and science of manipulating defensible metaphors."– ***G. Pask;***
7. "The art of creating equilibrium in a world of constraints and possibilities."– ***E. Glasersfeld;***
8. "The science and art of understanding."– ***H. Maturana;***
9. "A synthetic science of control, information and systems"– ***A.G. Butkovsky;***
10. "The art of interaction in dynamic networks." – ***R. Ascott.***

Almost all definitions involve the terms "control" and "system". Therefore, they are mutually noncontradictory and well consistent with the definition of cybernetics accepted by us. Consequently, Wiener's cybernetics has the following key terms: control, communication, system, information, feedback, black box, variety, homeostat.

Text 2. Historical Development of Cybernetics

1. Learn the vocabulary of the lesson

1. derive from
2. appear
3. government
4. concept

5. revive
6. seminal
7. elaborate
8. define
9. inspire
10. servomechanisms
11. contemporaneous
12. set out
13. distinguish
14. set goals
15. hosted by
16. noted
17. in terms of
18. artificial
19. cohere with
20. to broaden
21. to forge
22. contribution
23. loop
24. foresee
25. given (с учётом)
26. cyberneticists/cyberneticians
27. unavoidable
28. hypothesize
29. complexity
30. cellular automata
31. perceptron

2. Read and translate the text.

Derived from the Greek kybernetes the term "cybernetics" first appears in Antiquity with Plato, and in the 19th century with Ampère, who both saw it as the science of effective government. The concept was revived and elaborated by the mathematician Norbert Wiener in his seminal 1948 book, whose title defined it as "Cybernetics, or the study of control and communication in the animal and the machine".

Inspired by wartime and pre-war results in mechanical control systems Wiener set out to develop a general theory of organizational and control relations in systems. Information Theory, Control Theory and Control Systems Engineering have since developed into independent disciplines.

What distinguishes cybernetics is its emphasis on control and communication not only in engineered, artificial systems, but also in evolved, natural systems such as organisms and societies, which set their own goals, rather than being controlled by their creators.

Cybernetics as a specific field grew out of a series of interdisciplinary meetings held from 1944 to 1953 that brought together a number of noted post-war intellectuals, including Wiener, John von Neumann, Warren McCulloch, Claude Shannon, Heinz von Foerster, W. Ross Ashby, Gregory Bateson and Margaret Mead. Hosted by the Josiah Macy Jr. Foundation, these became known as the Macy Conferences on Cybernetics. From its original focus on machines and animals, cybernetics quickly broadened to encompass minds and social systems, thus recovering Plato's original focus on the control relations in society.

Through the 1950s, cybernetic thinkers came to cohere with the school of General Systems Theory (GST), founded at about the same time by Ludwig von Bertalanffy, as an attempt to build a unified science by uncovering the common principles that govern open, evolving systems. GST studies systems at all levels of generality, whereas Cybernetics focuses more specifically on goal-directed, functional systems which have some form of control relation.

Perhaps the most fundamental contribution of cybernetics is its explanation of purposiveness, or goal-directed behavior, an essential characteristic of mind and life, in terms of control and information. Negative feedback control loops which try to achieve and maintain goal states were seen as basic models for the autonomy characteristic of organisms: their behavior, while purposeful, is not strictly determined by either environmental influences or internal dynamical processes.

Thus cybernetics foresaw much current work in robotics and autonomous agents. Indeed, in the popular mind, "cyborgs" and "cybernetics" are just fancy terms for "robots" and "robotics". Given the technological advances of the post-war period, early cyberneticists were eager to explore the similarities between technological and biological systems. Armed with a theory of information, early digital circuits, and Boolean logic, it was unavoidable that they would hypothesize digital systems as models of brains, and information as the "mind" to the machine's "body". More generally, cybernetics had a crucial influence on the birth of various modern sciences: control theory, computer science, information theory, automata theory, artificial intelligence and artificial neural networks, cognitive science, computer modeling and simulation science, dynamical systems, and artificial life. Many concepts central to these fields, such as complexity, self-organization, self-reproduction, autonomy, networks, connectionism, and adaptation, were first explored by cyberneticians during the 1940's and 1950's. Examples include von Neumann's computer architectures, game theory, and cellular automata; Ashby's and von Foerster's analysis of self-organization; Braitenberg's autonomous robots; and McCulloch's artificial neural nets, perceptrons, and classifiers.

3. Answer the questions.

1. When did the term cybernetics first appear?
2. When was the concept revived and elaborated by the mathematician Norbert Wiener?
3. What made Wiener set out to develop a general theory of organizational and control relations in systems?
4. What distinguishes cybernetics?
5. Can you name some of the noted cybernetic thinkers?
6. Who founded the school of General Systems Theory (GST)?
7. Cybernetics focuses more specifically on goal-directed, functional systems which have some form of control relation, doesn't it?

4. Translate the following word combinations:

effective government, mechanical control systems, general theory, independent disciplines, engineered, artificial systems, natural systems, specific field, interdisciplinary meetings, noted post-war intellectuals, social systems, original focus, unified science, common principles, evolving systems, goal-directed, functional systems, fundamental contribution, goal-directed behavior, essential characteristic, negative feedback, basic models, environmental influences, internal dynamical processes, current work, autonomous agents, popular mind, fancy terms, technological advances, technological and biological systems, digital circuits, digital systems, crucial influence, various modern sciences, artificial intelligence, artificial neural networks, cognitive science, dynamical systems, artificial life.

5. Give 3 forms of the following irregular verbs:

See, set, grow, hold, bring, become, know, come, foresee, build.

Text 3. Cybernetics Today

1. Learn the vocabulary of the text

1. establish
2. relatively
3. research departments
4. devote
5. domain
6. intrinsic complexity
7. lack of
8. up-to-date

9. the ebb and flow of scientific fashions
10. apparent
11. similar
12. cause
13. maintain
14. coherence
15. broad
16. in the wake of
17. rapid growth
18. "spin-off" disciplines
19. to sap away
20. core ideas
21. the rebirth of neural networks
22. significance
23. significant
24. recent
25. subfield
26. to simulate
27. boundaries
28. the issues
29. ubiquitous
30. to pervade
31. albeit
32. shallow
33. explosive growth
34. software agents
35. far-reaching
36. encompassing conceptual framework
37. a semiotic theory of information

2. Read and translate the text.

In spite of its important historical role, cybernetics has not really become established as an autonomous discipline. Its practitioners are relatively few, and not very well organized. There are few research departments devoted to the domain, and even fewer academic programs. There are many reasons for this, including the intrinsic complexity and abstractness of the subject domain, the lack of up-to-date textbooks, the ebb and flow of scientific fashions, and the apparent overreaching of the second-order movement (чрезмерный охват направлений второго порядка). But the fact that the Systems Sciences (including General Systems Theory) are in a similar position indicates that the most important cause is the difficulty of maintaining the coherence of a broad, interdisciplinary field in the wake of the rapid growth of its more specialized and application-oriented "spin-off" disciplines, such as computer science, artificial intelligence, neural networks, and control engineering, which tended to sap away enthusiasm, funding and practitioners from the more theoretical mother field. Many of the core ideas of cybernetics have been assimilated by other disciplines, where they continue to influence scientific developments. Other important cybernetic principles seem to have been forgotten, though, only to be periodically rediscovered or reinvented in different domains. Some examples are the rebirth of neural networks, first invented by cyberneticists in the 1940's, in the late 1960's and again in the late 1980's; the rediscovery of the importance of autonomous interaction by robotics and AI in the 1990's; and the significance of positive feedback effects in complex systems, rediscovered by economists in the 1990's.

Perhaps the most significant recent development is the growth of the complex adaptive systems movement, which, in the work of authors such as John Holland, Stuart Kauffman and Brian Arthur and the subfield of artificial life, has used the power of modern computers to simulate and thus experiment with and develop many of the ideas of cybernetics. It thus seems to have taken over the cybernetics banner in its mathematical modelling of complex systems across disciplinary boundaries, however, while largely ignoring the issues of goal-directedness and control. More generally, as reflected by the ubiquitous prefix "cyber", the broad cybernetic philosophy that systems are defined by their abstract relations, functions, and information flows, rather than by their concrete material or components, is starting to pervade popular culture, albeit it in a still shallow manner, driven more by fashion than by deep understanding. This has been motivated primarily by the explosive growth of information-based technologies including automation, computers, the Internet, virtual reality, software agents, and robots. It seems likely that as the applications of these technologies become increasingly complex, far-reaching, and abstract, the need will again be felt for an encompassing conceptual framework, such as cybernetics, that can help users and designers alike to understand the meaning of these developments.

The sociocybernetics movement actively pursues a cybernetic understanding of social systems. The cybernetics-related programs on autopoiesis, systems dynamics and control theory also continue, with applications in management science and even psychological therapy. Scattered research centers, particularly in Central and Eastern Europe, are still devoted to specific technical applications, such as biological cybernetics, medical cybernetics, and engineering cybernetics, although they tend to keep closer contact with their field of application than with the broad theoretical development of cybernetics. General Information Theory has grown as the search for formal representations which are not based strictly on classical probability theory. There has also been significant progress in building a semiotic theory of information, where issues of the semantics and meaning of signals are at last being seriously considered. Finally, a number of authors are seriously questioning the limits of mechanism and formalism for interdisciplinary modeling in particular, and science in general. The issues here thus become what the ultimate limits on knowledge might be, especially as expressed in mathematical and computer-based models. What's at stake is whether it is possible, in principle, to construct models, whether formal or not, which will help us understand the full complexity of the world around us.

*Аутопоэзис (от греческих слов: αὐτός — «сам», ποίησις — «сотворение, производство») — термин, введённый в начале 1970-х годов чилийскими учёными Умберто Матураной и Франсиско Варелой. Означает самовоспроизводство, самостроение живых существ и образуемых ими систем (включая человека и общество). Организация таких систем порождает в качестве продукта только их самих без разделения на производителя и продукт.

3. Make up questions to the following sentences:

1. Cybernetics has not really become established as an autonomous discipline.
2. There are few research departments devoted to the domain, and even fewer academic programs.
3. Many of the core ideas of cybernetics have been assimilated by other disciplines, where they continue to influence scientific developments.
4. The broad cybernetic philosophy is starting to pervade popular culture, albeit it in a still shallow manner.
5. This has been motivated primarily by the explosive growth of information-based technologies including automation, computers, the Internet, virtual reality, software agents, and robots.

4. Insert prepositions:

In spite …, devote …, many reasons …, the lack …, to experiment …, to keep closer contact…, the search …, to be based….

5. Find the antonyms in the text:

Dependent, unpopular, insignificant, extrinsic, applied, primitive, disorganized, various, outdated, invisible/hidden.

TEXT 4. Subdivisions of cybernetics. Part 1.

1. Learn the vocabulary of the text:

1. the processing of information
2. subsequent
3. take shape
4. purpose
5. reveal

6. provide
7. intermediate
8. a research technique
9. furthermore
10. amenable
11. by comparison with
12. considerably
13. encompass
14. consist of
15. to steer towards
16. maintain
17. counteract disturbances
18. inherently
19. observer
20. emergence
21. disasters
22. distinguish

2. Read and translate the text.

The science of the control, communication and processing of information (literally "helmsmanship"). The subsequent development of cybernetics has been greatly influenced by the computer. At the beginning of the 1970s, cybernetics finally took shape as a science of mathematical-physical nature with the specific purpose of studying so-called cybernetic systems. Cybernetics systems are an abstraction from a specific (informational) point of view of the complex systems studied by a broad spectrum of natural, technical and social sciences (from their own specific standpoints). By revealing the general aspects in systems of such different nature, cybernetics provides a general and moreover new method of studying them. This is the so-called method of computer simulation, which is intermediate between the classical deductive method and the classical experimental method. Because of this, cybernetics, just like mathematics, can be used as a research technique in other sciences. Furthermore, the spectrum of problems amenable to investigation by cybernetic methods is, by comparison with classical (analytical) mathematical methods, considerably broader and encompasses practically all sciences.

Cybernetics is the science that studies the abstract principles of organization in complex systems. It is concerned not so much with what systems consist of, but how they function. Cybernetics focuses on how systems use information, models, and control actions to steer towards and maintain their goals, while counteracting various disturbances. Being inherently transdisciplinary, cybernetic reasoning can be applied to understand, model and design systems of any kind: physical, technological, biological, ecological, psychological, social, or any combination of those. Second-order cybernetics in particular studies the role of the (human) observer in the construction of models of systems and other observers.

Cybernetics studies systems of control as a concept, attempting to discover the basic principles underlying such things as artificial intelligence, computer vision, control systems, conversation theory, emergence, interactions of actors' theory, learning organization, robotics, second-order cybernetics, self-organization in cybernetics.

In biology: Cybernetics in biology is the study of cybernetic systems present in biological organisms, primarily focusing on how animals adapt to their environment, and how information in the form of genes is passed from generation to generation. There is also a secondary focus on combining artificial systems with biological systems. − Autopoiesis − Biocybernetics − Bioengineering − Bionics − Ecology − Heterostasis − Homeostasis − Medical cybernetics − Neuroscience − Practopoiesis − Synthetic biology − Systems biology.

In computer science: Computer science directly applies the concepts of cybernetics to the control of devices and the analysis of information. − Cellular automaton − Decision support systems − Design patterns − Robotics – Simulation.

In engineering: Cybernetics in engineering is used to analyze cascading failures and system accidents, in which the small errors and imperfections in a system can generate disasters. Other topics studied include: − Adaptive systems − Biomedical engineering − Engineering cybernetics − Ergonomics − Systems engineering.

In management:

− Autonomous agency theory − Entrepreneurial cybernetics − Management cybernetics − Operations research − Organizational cybernetics − Systems engineering − Viable system theory.

Economic cybernetics includes – mathematical programming – operation – research – mathematical economic models – econometrics – mathematical economics.

In mathematics:

Mathematics is regarded as an integral part of cybernetical studies and the contributions of the pioneering cyberneticians to the field are considerable. Norbert Wiener, the father of cybernetics was, of course, a distinguished mathematician in his own right.

Mathematical Cybernetics focuses on the factors of information, interaction of parts in systems, and the structure of systems. − Control theory − Dynamical system − Information theory − Systems theory

3. Answer the questions:

1. What is cybernetics?

2. What does it study?
3. What can cybernetic reasoning be applied for?
4. What is cybernetics in biology?
5. Does computer science directly apply the concepts of cybernetics?
6. What does Mathematical cybernetics focus on?
7. What does Economic cybernetics include?

4. Open the brackets (use Passive Voice):

1. Mathematics (regard) as an integral part of cybernetical studies.
2. The subsequent development of cybernetics greatly (influence) by the computer.
3. Cybernetics, just like mathematics, (can use) as a research technique in other sciences.
4. Cybernetic reasoning (can apply) to understand, model and design systems of any kind.
5. Information in the form of genes (pass) from generation to generation.
6. Cybernetics in engineering (use) to analyze cascading failures and system accidents, in which the small errors and imperfections in a system can generate disasters.
7. Mathematics (regard) as an integral part of cybernetical studies.

TEXT 4. Subdivisions of cybernetics. Part 2.

1. Learn the vocabulary of the text:

1. generic term
2. subject matter
3. extend
4. avoid obstacles
5. Racial Memory
6. cumulative adaptation
7. homeostasis
8. attempt
9. imperfections
10. smart mobs and riots
11. etiquette
12. memetics

2. Read and translate the text.

Cybernetics is an earlier but still-used generic term for many subject matters. These subjects also extend into many others areas of science, but are united in their study of control of systems.

Pure cybernetics

Pure cybernetics studies systems of control as a concept, attempting to discover the basic principles underlying such things as ASIMO uses sensors and intelligent algorithms to avoid obstacles and navigate stairs.

The development of computer science

–Artificial intelligence
–Robotics
–Computer Vision
–Control systems
–Emergence
–Learning organization
–New Cybernetics
–Second-order cybernetics
–Interactions of Actors Theory
–Conversation Theory

In biology

Cybernetics in biology is the study of cybernetic systems present in biological organisms, primarily focusing on how animals adapt to their environment, and how information in the form of genes is passed from generation to generation. (Note: this does not refer to the concept of Racial Memory but to the concept of cumulative adaptation to a particular niche, such as the case of the pepper moth having genes for both light and dark environments.) There is also a secondary focus on cyborgs.

–Bioengineering
–Biocybernetics
–Bionics
–Homeostasis
–Medical cybernetics
–Synthetic Biology
–Systems Biology

In complexity science

Complexity Science attempts to analyze the nature of complex systems, and the reasons behind their unusual properties.

–Complex Adaptive System
–Complex systems
–Complexity theory

In computer science

Computer science directly applies the concepts of cybernetics to the control of devices and the analysis of information.

–Robotics
–Decision support system
–Cellular automaton
–Simulation

In engineering

Cybernetics in engineering is used to analyze cascading failures and system accidents, in which the small errors and imperfections in a system can generate disasters. Other topics studied include:

–An artificial heart, example of a biomedical engineering
–Adaptive systems
–Engineering cybernetics
–Ergonomics
–Biomedical engineering
–Systems engineering

In management

–Entrepreneurial cybernetics
–Management cybernetics
–Organizational cybernetics
–Operations research
–Systems engineering

In mathematics

Mathematical Cybernetics focuses on the factors of information, interaction of parts in systems, and the structure of systems.

–Dynamical system
–Information theory
–Systems theory

In psychology

–Psycho-Cybernetics
–Systems psychology

In sociology

By examining group behavior through the lens of cybernetics, sociology seeks the reasons for such spontaneous events as smart mobs and riots, as well as how communities develop rules, such as etiquette, by consensus without formal discussion. Affect Control Theory (теория управления воздействием) explains role behavior, emotions, and labeling theory in terms of homeostatic maintenance of sentiments associated with cultural categories.

–Memetics
–Sociocybernetics.

TEXT 5. The relation of cybernetics to mathematics

1. Learn the vocabulary of the text:

1. restrict
2. solely
3. regard
4. a means
5. approach
6. differ
7. therefore
8. consist in
9. reduce
10. develop
11. suitable
12. necessitate
13. instead of
14. equivalence
15. suitable for
16. implementation
17. presuppose
18. feasibility
19. pose the problem
20. conceal
21. solution
22. constituents
23. pattern recognition
24. instruction and self-instruction
25. limbs
26. arise

2. Read and translate the text.

The relation of cybernetics to mathematics is not restricted solely to the use of mathematical methods in cybernetics. Mathematics and cybernetics also have common objects of investigation. Thus, algorithms, for example, which are an object of research in the mathematical theory of algorithms, can at the same time be regarded as cybernetic systems and are both a means and an object of study in cybernetics. However, the mathematical and cybernetic approaches differ greatly.

For mathematics, the algorithm stands out primarily as one of the fundamental concepts in the foundations of mathematics. The main problem, therefore, consists in the study of general properties of this concept, for which it is necessary to reduce its definition to a minimum number of simple basic concepts and operations. Cybernetics sets itself the problem of developing suitable methods in practice for the synthesis of concrete systems, among them algorithms. This practical approach necessitates the development of problem- and procedure-oriented algorithmic languages that are suitable for applications. Instead of being concerned with the possibility of establishing equivalence of certain classes of algorithms, which is characteristic for mathematics, cybernetics is primarily interested in the creation of apparatus suitable for the actual implementation of equivalent transformations of algorithms. Instead of the simplest form for the representation of information, as words over an abstract alphabet, cybernetics studies the complex data structures that are necessary for the effective realization of algorithms on a computer. The examples mentioned characterize fairly clearly the special nature of the cybernetic approach to the study of its mathematical objects.

The same difference in approach can be detected in the cybernetic study of other mathematical objects (abstract automata, logical networks, etc.).

Of special interest from the point of view of the interrelation between cybernetics and mathematics is the approach of each of them to the apparatus of classical mathematical logic. The mathematical point of view presupposes a maximal simplification of the axiom system and deduction rules, without which an effective analysis of its general properties and of the feasibility of logical calculi is impossible. Cybernetics, by posing the problem of the automation of deductive constructions by practical means, started to develop the language of practical mathematical logic. The language relates to that of mathematical logic and, as a modern programming language to the language of Post algorithms or Markov's normal algorithms. The deduction rules in this language (in their interaction with actual meaningful mathematical texts) have a demonstrative power which is no less than that concealed in the word "obvious" in current mathematical monographs.

The automation of deductive constructions is one of the most important parts of this branch of cybernetics, and is termed "artificial intelligence". The natural human intellect (the brain together with organs that receive and deliver external information) is one of the most interesting and complex cybernetic systems. The problem concerning how man thinks has been and continues to be a most interesting and fascinating scientific question. Cybernetics approaches its solution both from the theoretical as well as the practical angle. It is a question of automation (full or partial with human intervention) of the various aspects of human intellectual activity and, in the end, of intelligence as a whole.

Besides the automation of logical thought (deductive constructions), important constituents of the problem of "artificial intelligence" are problems of pattern recognition (primarily visual and aural), operations in natural human languages (recognition of the meaning of sentences and expressions, maintaining a dialogue, etc.), problems of instruction and self-instruction, etc. Of great importance are the problems of the study and synthesis of effective algorithms for the control of the movements of the artificial limbs of humanoid robots, the synthesis of an artificial voice, voice control, etc. A special group of problems arises in the study of purposeful behaviour, the methods of choosing goals, sub-goals and schemes for their achievement. A practical approach to the creation of artificial intelligence lies in the creation of dialogue (human—machine) systems, that increase the productivity of the human in various areas of intellectual activity (logical deduction, translation from one natural language to another, chess playing, etc.).

As these become perfected, the role played by the computer in collaborative work will continually increase right up to a full automation of the corresponding process.

https://encyclopediaofmath.org/wiki/Cybernetics

3. Make up questions:

1. Mathematics and cybernetics also have common objects of investigation. (General question).
2. The mathematical and cybernetic approaches differ greatly. (Special question).
3. Cybernetics sets itself the problem of developing suitable methods in practice for the synthesis of concrete systems. (Disjunctive question).
4. Cybernetics is interested in the creation of apparatus suitable for the actual implementation of equivalent transformations of algorithms. (Special question).
5. Cybernetics, by posing the problem of the automation of deductive constructions by practical means, started to develop the language of practical mathematical logic. (Alternative question).

4. True or false:

1. Mathematics and cybernetics have common objects of investigation.

2. Algorithms being an object of research in the mathematical theory of algorithms can be regarded as cybernetic systems.

3. Mathematical and cybernetic approaches don't differ greatly.

4. The natural human intellect is one of the most interesting and complex cybernetic systems.

5. A practical approach to the creation of artificial intelligence lies in the creation of dialogue (human-machine) systems.

TEXT 6. Interrelation between cybernetics and modern computer technology

1. Learn the vocabulary of the lesson:

1. stand out
2. investigation
3. solve
4. enter into
5. fall within
6. include
7. constitute
8. branch
9. it is worth
10. bear in mind
11. distinction
12. subject matter
13. encounter
14. differ
15. property
16. relate

1. Read and translate the text.

Of great importance is the question of the interrelation between cybernetics and modern computer technology. First, the computer is the basic instrument of investigation for cybernetics, and secondly, it stands out, being the most complex of cybernetic systems, as an important object of investigation in cybernetics. Of course there are many problems which have to be solved when actually designing a computer that do not enter into cybernetics. However, the questions of the architecture of computers and computer systems, the organization of the control by the computing process (including the organization of data bases) fall within the competence of cybernetics and constitute one of its more important branches. This is worth bearing in mind, since it shows up the distinction in the subject matter of cybernetics, where questions of architecture and control of computing systems relate specifically to "computer science" and are not included in cybernetics.

Cybernetic systems are encountered in practically all areas of knowledge. Systems, differing in certain specific properties, can be studied by cybernetic methods specially adapted to the systems of the corresponding classes. In this way a deeper study of them becomes possible. Thus, specialized applied branches of cybernetics have arisen and continue to be developed, such as technological cybernetics, economic cybernetics, biological cybernetics, medical cybernetics, and military cybernetics.

The application of mathematical and cybernetic methods in linguistics has given rise to the topic called mathematical linguistics. This science is directly related to linguistic problems of artificial intelligence.

2. Insert suitable prepositions:

1. Cybernetic systems are encountered … practically all areas of knowledge.
2. Of great importance is the question … the interrelation between cybernetics and modern computer technology.
3. This is worth bearing … mind.
4. The application … mathematical and cybernetic methods … linguistics has given rise to the topic called mathematical linguistics.
5. The questions of the architecture of computers and computer systems, the organization of the control by the computing process fall … the competence of cybernetics.

3. Identify the following verb forms:

are encountered, has given, constitute, stands out, are not included, can be studied, have arisen

TEXT 7. Cybernetics as a science

1. Learn the vocabulary of the lesson:

1. determine
2. divide into
3. restrict
4. certain
5. set
6. applicable
7. serve
8. description
9. generalize
10. research
11. split
12. intersect
13. cause
14. condition
15. appearance
16. survival
17. sufficient
18. origin/source sciences
19. numerous
20. partial results

21. scientific community
22. satisfy
23. long-term cooperation
24. alongside with
25. ensure
26. obtain
27. substantiated applicability
28. give birth to
29. elucidation
30. forecast

2. Read and translate the text:

Any science is determined by its “subject” (problem domain) and “method” (an integrated set of methods). Therefore, sciences can be divided into:

– *subject-oriented sciences* studying a certain subject by different methods (e.g., physics, biology, sociology);

– *method-oriented sciences* (in the restricted sense, the so-called model-based sciences) developing a certain set of methods applicable to different subjects; for instance, a classical example is applied mathematics: the apparatus and methods of its branches (differential equations, game theory, etc.) serve for description and analysis of different-nature systems;

– *synthetic sciences* (“metasciences”) mostly developing and/or generalizing methods of certain sciences in application to subjects of these and/or other sciences (e.g., operations research, systems analysis, cybernetics). With the course of time, synthetic sciences find or generate their “own” subjects and methods. As sciences of any types develop, their subjects and methods are split and intersected by each other, causing further differentiation of sciences.

The following conditions guarantee the appearance (1 and 2) and survival (3) of synthetic sciences:

1) a sufficient development level of origin/source sciences;

2) numerous analogies (and then generalizations) among partial results of source sciences;

3) rather easy and fast generation/accumulation of nontrivial theoretical and applied results and their popularization, within the scientific community and everymen.

Speaking about cybernetics, the first and second conditions had been satisfied by the middle of the 1940’s. And the long-term cooperation between N. Wiener and biologists, alongside with his wide and deep professional interests (recall Wiener processes, BanachWiener spaces, the Wiener-Hopf equations) ensured “subjective” satisfaction of these conditions. In 1948 integration of results obtained by different sciences and their substantiated applicability to different subjects gave birth to a new synthetic science known as Wiener's cybernetics.

A science as a system of knowledge has the following epistemological functions:

– descriptive (phenomenological) function, i.e., acquisition and accumulation of data and facts. Any science starts from this function, viz., answering to the question "What is the structure of the world?", as any science can be based on very many facts. From this viewpoint, cybernetics as a synthetic science mostly employs the results of its components (source sciences);

– explanatory (explicative) function, i.e., elucidation of phenomena and processes, their internal mechanisms. Here the question to-be answered is "Why does the world is exactly this?". In this function, cybernetics plays a more visible role: even analogies may have powerful elucidation;

– generalizing function, i.e., formulation of laws and regularities systematizing and absorbing numerous fragmented phenomena and facts (the associated question is "What are the common features of ...?"). Perhaps, this is the main function of cybernetics, since generalizations (in the form of laws, regularities, models, research approaches) comprise the framework of its results;

– predictive (prognostic) function, i.e., scientific knowledge allows predicting new processes and phenomena (this function answers the question "What and why will happen?"). Efficient forecasting is possible using substantiated analogies and constructive generalizations within synthetic science cybernetics;

– prescriptive (normative) function, i.e., scientific knowledge allows organizing activity with certain goals (the corresponding question is "What and how should be done for goal achievement?").

– normative function has a close connection with solution of control problems, an important subject of cybernetics.

3. Agree or disagree:

1. Cybernetics as a synthetic science mostly employs the results of source sciences.
2. The apparatus and methods of applied math branches (differential equations, game theory, etc.) serve for description and analysis of different-nature systems.
3. As sciences of any types develop, their subjects and methods are split and intersected by each other, causing further differentiation of sciences.
4. Any science starts from this function, viz., answering to the question "What is the structure of the world?"
5. Generalizing function is the main function of cybernetics, since generalizations (in the form of laws, regularities, models, research approaches) comprise the framework of its results.

4. Participle I or Participle II?

1. *Speaking about* cybernetics, the first and second conditions had been satisfied by the middle of the 1940's.
2. Efficient forecasting is possible *using substantiated* analogies and constructive generalizations within synthetic science cybernetics.
3. *Subject-oriented* sciences studying a certain subject by different methods.
4. Subjects and methods are split and intersected by each other, *causing* further differentiation of sciences.
5. In 1948 integration of results *obtained* by different sciences and their substantiated applicability to different subjects gave birth to a new synthetic science known as Wiener's cybernetics.

UNIT 2. COMPUTER SCIENCE

Text 1. The development of computer science

1. Learn the vocabulary of the lesson:

1. to process information
2. to trace the roots
3. to propose
4. the advent of
5. to distinguish
6. to separate
7. to sprout
8. to enable
9. efficient
10. approach
11. to perform various calculations
12. to overlap
13. numerical analysis
14. the accuracy and precision of calculations
15. to expand
16. to broaden
17. to include
18. to simplify
19. artificial languages
20. to provide a useful interface
21. computer scientists

22. applications and computer designs
23. to explore
24. computer chip manufacturers
25. the electronic circuitry
26. to reduce the cost
27. to increase the processing speed
28. to result in
29. an explosion
30. the use of computer applications
31. an effort
32. to drive the technological advances in the computing industry
33. to reach the public
34. to derive
35. complex, reliable, and powerful computers
36. to exchange vast amounts of information
37. to behave intelligently
38. an increasingly integral part of modern society
39. strive to solve new problems
40. current problems
41. the goal
42. to range from…to…
43. speculative research into technologies
44. to be viable
45. the improved use of information
46. hardware and software
47. a theory-driven approach
48. software engineering tools
49. to evaluate
50. an artificial neural network
51. the outcome of experiments
52. in advance

2. Read and translate the text.

Introduction

Computer Science, study of the theory, experimentation, and engineering that form the basis for the design and use of computers-devices that automatically process information. Computer science traces its roots to work done by English mathematician Charles Babbage, who first proposed a programmable mechanical calculator in 1837. Until the advent of electronic digital computers in the 1940s, computer science was not generally distinguished as being separate from mathematics and engineering. Since then it has sprouted numerous branches of research that are unique to the discipline.

The development of computer science

Early work in the field of computer science during the late 1940s and early 1950s focused on automating the process of making calculations for use in science and engineering. Scientists and engineers developed theoretical models of computation that enabled them to analyze how efficient different approaches were in performing various calculations. Computer science overlapped considerably during this time with the branch of mathematics known as numerical analysis, which examines the accuracy and precision of calculations.

As the use of computers expanded between the 1950s and the 1970s, the focus of computer science broadened to include simplifying the use of computers through programming languages-artificial languages used to program computers, and operating systems-computer programs that provide a useful interface between a computer and a user. During this time, computer scientists were also experimenting with new applications and computer designs, creating the first computer networks, and exploring relationships between computation and thought.

In the 1970s, computer chip manufacturers began to mass produce microprocessors-the electronic circuitry that serves as the main information processing center in a computer. This new technology revolutionized the computer industry by dramatically reducing the cost of building computers and greatly increasing their processing speed. The microprocessor made possible the advent of the personal computer, which resulted in an explosion in the use of computer applications. Between the early 1970s and 1980s, computer science rapidly expanded in an effort to develop new applications for personal computers and to drive the technological advances in the computing industry. Much of the earlier research that had been done began to reach the public through personal computers, which derived most of their early software from existing concepts and systems.

Computer scientists continue to expand the frontiers of computer and information systems by pioneering the designs of more complex, reliable, and powerful computers; enabling networks of computers to efficiently exchange vast amounts of information; and seeking ways to make computers behave intelligently. As computers become an increasingly integral part of modern society, computer scientists strive to solve new problems and invent better methods of solving current problems.

The goals of computer science range from finding ways to better educate people in the use of existing computers to highly speculative research into technologies and approaches that may not be viable for decades. Underlying all of these specific goals is the desire to better the human condition today and in the future through the improved use of information.

Theory and experiment

Computer science is a combination of theory, engineering, and experimentation. In some cases, a computer scientist develops a theory, then engineers a combination of computer hardware and software based on that theory, and experimentally tests it. An example of such a theory-driven approach is the development of new software engineering tools that are then evaluated in actual use. In other cases, experimentation may result in new theory, such as the discovery that an artificial neural network exhibits behavior similar to neurons in the brain, leading to a new theory in neurophysiology.

It might seem that the predictable nature of computers makes experimentation unnecessary because the outcome of experiments should be known in advance. But when computer systems and their interactions with the natural world become sufficiently complex, unforeseen behaviors can result. Experimentation and the traditional scientific method are thus key parts of computer science.

3. Read the text about Charles Babbage in Russian and retell it in English.

Чарлз Бэббидж родился 26 декабря 1791 года в Лондоне в семье банкира Бенджамина Бэббиджа и Элизабет Тип (англ. *Teape*). В детстве у Чарльза было очень слабое здоровье. В 8 лет его отправили в частную школу в Альфингтоне на воспитание священнику. На тот момент его отец уже был достаточно обеспечен, чтобы позволить обучение Чарльза в частной школе. Бенджамин Бэббидж попросил священника не давать Чарльзу сильных учебных нагрузок из-за слабого здоровья.

После школы в Альфингтоне Чарлз был отправлен в академию в Энфилде, где по существу и началось его настоящее обучение. Именно там Бэббидж начал проявлять интерес к математике, чему поспособствовала большая библиотека в академии.

В 1810 году Бэббидж поступил в Тринити-колледж в Кембридже. Однако основам математики он обучался самостоятельно по книжкам. Он тщательно изучал труды Ньютона, Лейбница, Лагранжа, Лакруа, Эйлера и других математиков академий Санкт-Петербурга, Берлина и Парижа. Бэббидж очень быстро обогнал своих преподавателей по знаниям и был сильно разочарован уровнем преподавания математики в Кембридже. Более того, он заметил, что Британия в целом заметно отстала от континентальных стран по уровню математической подготовки.

В связи с этим он решил создать общество, целью которого являлось внесение современной европейской математики в Кембриджский университет. В 1812 году Чарлз Бэббидж, его друзья, Джон Гершель (John Herschel) и Джордж Пикок (George Peacock) и ещё несколько молодых математиков основали «Аналитическое общество». Они стали проводить собрания. Обсуждать различные вопросы, связанные с математикой. Начали публиковать свои труды. Например, в 1816 году они опубликовали переведённый ими на английский язык «Трактат по дифференциальному и интегральному исчислению» французского математика Лакруа, а в 1820 году опубликовали два тома примеров, дополняющих этот трактат. Аналитическое общество своей активностью инициировало реформу математического образования вначале в Кембридже, а затем и в других университетах Британии.

В 1812 году Бэббидж перешёл в колледж Св. Петра (Peterhouse), а в 1814 году он получил степень бакалавра.

В 1816 году он стал членом Королевского Общества Лондона. К тому времени им было написано несколько больших научных статей в разных математических дисциплинах. В 1820 году он стал членом Королевского общества Эдинбурга и Королевского астрономического общества. В 1827 году он похоронил отца, жену и двоих детей. В 1827 году он стал профессором математических наук в Кембридже и занимал этот пост в течение 12 лет. После того, как он покинул этот пост, он большую часть своего времени посвятил делу его жизни – разработке вычислительных машин.

Последние годы жизни Бэббидж посвятил философии и политической экономии.

Чарлз Бэббидж умер в возрасте 79 лет 18 октября 1871 года. Похоронен на кладбище Кенсал Грин (англ. *Kensal Green Cemetery*) в Лондоне.

Text 2. Major branches of computer science

1. Learn the vocabulary of the lesson:

1. software development
2. computer architecture (hardware)
3. human- computer interfacing
4. artificial intelligence
5. the best types of programming languages and algorithms
6. to store and retrieve information
7. program performance
8. to sacrifice
9. for the sake of

10. a limited amount of memory
11. to limit the number of features
12. to require
13. to supply
14. to facilitate
15. robust
16. the software life cycle
17. implementation
18. program maintenance (сопровождение, администрирование программы)
19. programming environments
20. to improve the development process
21. a precise step-by-step procedure
22. matrix multiplication
23. data values
24. lists, arrays, records, stacks, queues, trees
25. determining the inherent (исходный, внутренний) efficiency of algorithms
26. computability theory
27. databases and information retrieval
28. to access databases
29. to prevent access by unauthorized users
30. to improve access speed
31. to compress the data
32. to update the data simultaneously
33. reduce access speed
34. information retrieval
35. control the computer's input and output devices

2. Read and translate the text.

Computer science can be divided into four main fields: software development, computer architecture (hardware), human- computer interfacing (the design of the most efficient ways for humans to use computers), and artificial intelligence (the attempt to make computers behave intelligently). Software development is concerned with creating computer programs that perform efficiently. Computer architecture is concerned with developing optimal hardware for specific computational needs. The areas of artificial intelligence (AI) and human-computer interfacing often involve the development of both software and hardware to solve specific problems.

Software development

In developing computer software, computer scientists and engineers study various areas and techniques of software design, such as the best types of programming languages and algorithms to use in specific programs, how to efficiently store and retrieve information, and the computational limits of certain software-computer combinations. Software designers must consider many factors when developing a program. Often, program performance (производительность программы) in one area must be sacrificed for the sake of the general performance of the software. For instance, since computers have only a limited amount of memory, software designers must limit the number of features they include in a program so that it will not require more memory than the system it is designed for can supply.

Software engineering is an area of software development in which computer scientists and engineers study methods and tools that facilitate the efficient development of correct, reliable, and robust (надежные, устойчивые к ошибкам) computer programs. Research in this branch of computer science considers all the phases of the software life cycle, which begins with a formal problem specification, and progresses to the design of a solution, its implementation as a program, testing of the program, and program maintenance. Software engineers develop software tools and collections of tools called programming environments to improve the development process. For example, tools can help to manage the many components of a large program that is being written by a team of programmers.

Algorithms and data structures are the building blocks of computer programs. An algorithm is a precise step-by-step procedure for solving a problem within a finite time and using a finite amount of memory. Common algorithms include searching a collection of data, sorting data, and numerical operations such as matrix multiplication. Data structures are patterns for organizing information, and often represent relationships between data values. Some common data structures are called lists, arrays, records, stacks, queues, and trees.

Computer scientists continue to develop new algorithms and data structures to solve new problems and improve the efficiency of existing programs. One area of theoretical research is called algorithmic complexity. Computer scientists in this field seek to develop techniques for determining the inherent efficiency of algorithms with respect to one another. Another area of theoretical research called computability theory seeks to identify the inherent limits of computation.

Software engineers use programming languages to communicate algorithms to a computer. Natural languages such as English are ambiguous-meaning that their grammatical structure and vocabulary can be interpreted in multiple ways-so they are not suited for programming. Instead, simple and unambiguous artificial languages are used. Computer scientists study ways of making programming languages more expressive, thereby simplifying programming and reducing errors.

A program written in a programming language must be translated into machine language (the actual instructions that the computer follows). Computer scientists also develop better translation algorithms that produce more efficient machine language programs.

Databases and information retrieval are related fields of research. A database is an organized collection of information stored in a computer, such as a company's customer account data. Computer scientists attempt to make it easier for users to access databases, prevent access by unauthorized users, and improve access speed. They are also interested in developing techniques to compress the data, so that more can be stored in the same amount of memory. Databases are sometimes distributed over multiple computers that update the data simultaneously, which can lead to inconsistency in the stored information. To address this problem, computer scientists also study ways of preventing inconsistency without reducing access speed.

Information retrieval is concerned with locating data in collections that are not clearly organized, such as a file of newspaper articles. Computer scientists develop algorithms for creating indexes of the data. Once the information is indexed, techniques developed for databases can be used to organize it. *Data mining* is a closely related field in which a large body of information is analyzed to identify patterns. For example, mining the sales records from a grocery store could identify shopping patterns to help guide the store in stocking its shelves more effectively.

Operating systems are programs that control the overall functioning of a computer. They provide the user interface, place programs into the computer's memory and cause it to execute them, control the computer's input and output devices, manage the computer's resources such as its disk space, protect the computer from unauthorized use, and keep stored data secure. Computer scientists are interested in making operating systems easier to use, more secure, and more efficient by developing new user interface designs, designing new mechanisms that allow data to be shared while preventing access to sensitive data, and developing algorithms that make more effective use of the computer's time and memory.

The study of numerical computation involves the development of algorithms for calculations, often on large sets of data or with high precision. Because many of these computations may take days or months to execute, computer scientists are interested in making the calculations as efficient as possible. They also explore ways to increase the numerical precision of computations, which can have such effects as improving the accuracy of a weather forecast. The goals of improving efficiency and precision often conflict, with greater efficiency being obtained at the cost of precision and vice versa.

Symbolic computation involves programs that manipulate nonnumeric symbols, such as characters, words, drawings, algebraic expressions, encrypted data (data coded to prevent unauthorized access), and the parts of data structures that represent relationships between values. One unifying property of symbolic programs is that they often lack the regular patterns of processing found in many numerical computations. Such irregularities (нарушения, несоответствия) present computer scientists with special challenges in creating theoretical models of a program's efficiency, in translating it into an efficient machine language program, and in specifying and testing its correct behavior.

3. Open the brackets:

1. Computer science (can divide) into four main fields.
2. Often, program performance in one area (must sacrifice) for the sake of the general performance of the software.
3. Research in this branch of computer science (consider) all the phases of the software life cycle.
4. Some common data structures (call) lists, arrays, records, stacks, queues, and trees.
5. One area of theoretical research (call) algorithmic complexity.
6. Instead, simple and unambiguous artificial languages (use).
7. A program written in a programming language (must translate) into machine language.
8. Databases sometimes (distribute) over multiple computers that update the data simultaneously.
9. Computer scientists (develop) algorithms for creating indexes of the data.
10. Data mining is a closely related field in which a large body of information (analyze) to identify patterns.

Text 3. Architecture (computer science)

1. Learn the vocabulary of the lesson:

1. architecture
2. referring to
3. the design of system software
4. the combination of hardware and basic software
5. microprocessors
6. circuits
7. application programs
8. spreadsheets
9. word processing

10. to perform a task
11. to make the system run
12. make up the system's hardware
13. the arithmetic/logic unit
14. control unit
15. memory
16. input
17. output
18. to compare numerical values
19. the computer's circuitry
20. the central processing unit (CPU)
21. to receive and send data
22. to display graphics
23. to conserve battery power in a laptop computer
24. invisible to the user
25. to use different hardware architectures
26. to carry out an instruction
27. to carry out complex instructions
28. to decode the instructions into electronic signals
29. to fetch the data
30. to save the result
31. to decrease
32. to retrieve additional instructions
33. to eliminate
34. to increase overall performance
35. to provide special instruction sets
36. to expand
37. closed architectures
38. a ring configuration
39. a bus configuration
40. a star configuration
41. supplies instructions and data

2. Read and translate the text "Architecture (computer science)"

Introduction

Architecture (computer science), a general term referring to the structure of all or part of a computer system. The term also covers the design of system software, such as the operating system (the program that controls the computer), as well as referring to the combination of hardware and basic software that links the machines on a computer network. Computer architecture refers to an entire structure and to the details needed to make it functional. Thus, computer architecture covers computer systems, microprocessors, circuits, and system programs. Typically, the term does not refer to application programs, such as spreadsheets or word processing, which are required to perform a task but not to make the system run.

Design elements

In designing a computer system, architects consider five major elements that make up the system's hardware: the arithmetic/logic unit, control unit, memory, input, and output. The arithmetic/logic unit performs arithmetic and compares numerical values. The control unit directs the operation of the computer by taking the user instructions and transforming them into electrical signals that the computer's circuitry can understand. The combination of the arithmetic/logic unit and the control unit is called the central processing unit (CPU). The memory stores instructions and data. The input and output sections allow the computer to receive and send data, respectively.

Different hardware architectures are required because of the specialized needs of systems and users. One user may need a system to display graphics extremely fast, while another system may have to be optimized for searching a database or conserving battery power in a laptop computer.

In addition to the hardware design, the architects must consider what software programs will operate the system. Software, such as programming languages and operating systems, makes the details of the hardware architecture invisible to the user. For example, computers that use the C programming language or a UNIX operating system may appear the same from the user's viewpoint, although they use different hardware architectures.

Processing architecture

When a computer carries out an instruction, it proceeds through five steps. First, the control unit retrieves the instruction from memory—for example, an instruction to add two numbers. Second, the control unit decodes the instructions into electronic signals that control the computer. Third, the control unit fetches the data (the two numbers). Fourth, the arithmetic/logic unit performs the specific operation (the addition of the two numbers). Fifth, the control unit saves the result (the sum of the two numbers).

Early computers used only simple instructions because the cost of electronics capable of carrying out complex instructions was high. As this cost decreased in the 1960s, more complicated instructions became possible. Complex instructions (single instructions that specify multiple operations) can save time because they make it unnecessary for the computer to retrieve additional instructions. For example, if seven operations are combined in one instruction, then six of the steps that fetch instructions are eliminated and the computer spends less time processing that operation. Computers that combine several instructions into a single operation are called complex instruction set computers (CISC).

However, most programs do not often use complex instructions, but consist mostly of simple instructions. When these simple instructions are run on CISC architectures they slow down processing because each instruction-whether simple or complex-takes longer to decode in a CISC design. An alternative strategy is to return to designs that use only simple, single-operation instruction sets and make the most frequently used operations faster in order to increase overall performance. Computers that follow this design are called reduced instruction set computers (RISC).

RISC designs are especially fast at the numerical computations required in science, graphics, and engineering applications. CISC designs are commonly used for nonnumerical computations because they provide special instruction sets for handling character data, such as text in a word processing program. Specialized CISC architectures, called digital signal processors, exist to accelerate processing of digitized audio and video signals.

Open and closed architectures

The CPU of a computer is connected to memory and to the outside world by means of either an open or a closed architecture. An open architecture can be expanded after the system has been built, usually by adding extra circuitry, such as a new microprocessor computer chip connected to the main system. The specifications of the circuitry are made public, allowing other companies to manufacture these expansion products.

Closed architectures are usually employed in specialized computers that will not require expansion-for example, computers that control microwave ovens. Some computer manufacturers have used closed architectures so that their customers can purchase expansion circuitry only from them. This allows the manufacturer to charge more and reduces the options for the consumer.

Network architecture

Computers communicate with other computers via networks. The simplest network is a direct connection between two computers. However, computers can also be connected over large networks, allowing users to exchange data, communicate via electronic mail, and share resources such as printers.

Computers can be connected in several ways. In a ring configuration, data are transmitted along the ring and each computer in the ring examines this data to determine if it is the intended recipient. If the data are not intended for a particular computer, the computer passes the data to the next computer in the ring. This process is repeated until the data arrive at their intended destination. A ring network allows multiple messages to be carried simultaneously, but since each message is checked by each computer, data transmission is slowed.

In a bus configuration, computers are connected through a single set of wires, called a bus. One computer sends data to another by broadcasting the address of the receiver and the data over the bus. All the computers in the network look at the address simultaneously, and the intended recipient accepts the data. A bus network, unlike a ring network, allows data to be sent directly from one computer to another. However, only one computer at a time can transmit data. The others must wait to send their messages.

In a star configuration, computers are linked to a central computer called a hub. A computer sends the address of the receiver and the data to the hub, which then links the sending and receiving computers directly. A star network allows multiple messages to be sent simultaneously, but it is more costly because it uses an additional computer, the hub, to direct the data.

Recent advances

One problem in computer architecture is caused by the difference between the speed of the CPU and the speed at which memory supplies instructions and data. Modern CPUs can process instructions in 3 nanoseconds (3 billionths of a second). A typical memory access, however, takes 100 nanoseconds and each instruction may require multiple accesses. To compensate for this disparity, new computer chips have been designed that contain small memories, called caches, located near the CPU. Because of their proximity to the CPU and their small size, caches can supply instructions and data faster than normal memory. Cache memory stores the most frequently used instructions and data and can greatly increase efficiency.

Although a larger cache memory can hold more data, it also becomes slower. To compensate, computer architects employ designs with multiple caches. The design places the smallest and fastest cache nearest the CPU and locates a second larger and slower cache farther away. This arrangement allows the CPU to operate on the most frequently accessed instructions and data at top speed and to slow down only slightly when accessing the secondary cache. Using separate caches for instructions and data also allows the CPU to retrieve an instruction and data simultaneously.

Another strategy to increase speed and efficiency is the use of multiple arithmetic/logic units for simultaneous operations, called superscalar execution. In this design, instructions are acquired in groups. The control unit examines each group to see if it contains instructions that can be performed together. Some designs execute as many as six operations simultaneously. It is rare, however, to have this many instructions run together, so on average the CPU does not achieve a six-fold increase in performance.

Multiple computers are sometimes combined into single systems called parallel processors. When a machine has more than one thousand arithmetic/logic units, it is said to be massively parallel. Such machines are used primarily for numerically intensive scientific and engineering computation. Parallel machines containing as many as sixteen thousand computers have been constructed.

3. Make special questions to the sentences:

1. Computer architecture is the design and analysis of new computer systems.
2. Computer architects study ways of improving computers by increasing their speed, storage capacity, and reliability, and by reducing their cost and power consumption.
3. Computer architects develop both software and hardware models to analyze the performance of existing and proposed computer designs, then use this analysis to guide development of new computers.
4. They are often involved with the engineering of a new computer because the accuracy of their models depends on the design of the computer's circuitry.
5. Many computer architects are interested in developing computers that are specialized for particular applications such as image processing, signal processing, or the control of mechanical systems.
6. The optimization of computer architecture to specific tasks often yields higher performance, lower cost, or both.

Text 4. Artificial intelligence, robotics, human-computer interfacing

1. Learn the vocabulary of the text:

1. Artificial Intelligence (AI)
2. to mimic human intelligence and sensory processing ability
3. model human behavior
4. to improve our understanding of intelligence
5. machine learning
6. inference
7. cognition
8. knowledge representation
9. problem solving
10. case-based reasoning
11. natural language understanding
12. speech recognition
13. computer vision
14. artificial neural networks
15. the use of *heuristics*

16. computer controlled mechanical devices
17. automated factory assembly lines
18. to relieve humans from tedious, repetitive, or dangerous tasks
19. to model the robot's physical properties
20. to simplify the creation of control programs
21. a human-computer interface
22. to improving computer access for people with disabilities
23. to simplify program use
24. to develop three-dimensional input and output devices for virtual reality
25. to improve handwriting and speech recognition
26. to develop heads-up displays for aircraft
27. psychology
28. neurophysiology
29. linguistics
30. a highly interdisciplinary field of study

2. Read and translate the text:

Artificial intelligence (AI) research seeks to enable computers and machines to mimic human intelligence and sensory processing ability, and model human behavior with computers to improve our understanding of intelligence. The many branches of AI research include machine learning, inference, cognition, knowledge representation, problem solving, case-based reasoning, natural language understanding, speech recognition, computer vision, and artificial neural networks.

A key technique developed in the study of artificial intelligence is to specify a problem as a set of states, some of which are solutions, and then search for solution states. For example, in chess, each move creates a new state. If a computer searched the states resulting from all possible sequences of moves, it could identify those that win the game. However, the number of states associated with many problems (such as the possible number of moves needed to win a chess game) is so vast that exhaustively searching them is impractical. The search process can be improved through the use of *heuristics*-rules that are specific to a given problem and can therefore help guide the search. For example, a chess heuristic might indicate that when a move results in checkmate, there is no point in examining alternate moves.

Robotics

Another area of computer science that has found wide practical use is *robotics* – the design and development of computer controlled mechanical devices. Robots range in complexity from toys to automated factory assembly lines, and relieve humans from tedious, repetitive, or dangerous tasks. Robots are also employed where requirements of speed, precision, consistency, or cleanliness exceed what humans can accomplish. Roboticists-scientists involved in the field of robotics-study the many aspects of controlling robots. These aspects include modeling the robot's physical properties, modeling its environment, planning its actions, directing its mechanisms efficiently, using sensors to provide feedback to the controlling program, and ensuring the safety of its behavior. They also study ways of simplifying the creation of control programs. One area of research seeks to provide robots with more of the dexterity and adaptability of humans, and is closely associated with AI.

Human-computer interfacing

Human-computer interfaces provide the means for people to use computers. An example of a human-computer interface is the keyboard, which lets humans enter commands into a computer and enter text into a specific application. The diversity of research into human-computer interfacing corresponds to the diversity of computer users and applications. However, a unifying theme is the development of better interfaces and experimental evaluation of their effectiveness. Examples include improving computer access for people with disabilities, simplifying program use, developing three-dimensional input and output devices for virtual reality, improving handwriting and speech recognition, and developing heads-up displays for aircraft instruments in which critical information such as speed, altitude, and heading are displayed on a screen in front of the pilot's window. One area of research, called visualization, is concerned with graphically presenting large amounts of data so that people can comprehend its key properties.

Connection of computer science to other disciplines

Because computer science grew out of mathematics and electrical engineering, it retains many close connections to those disciplines. Theoretical computer science draws many of its approaches from mathematics and logic. Research in numerical computation overlaps with mathematics research in numerical analysis. Computer architects work closely with the electrical engineers who design the circuits of a computer.

Beyond these historical connections, there are strong ties between AI research and psychology, neurophysiology, and linguistics. Human-computer interface research also has connections with psychology. Roboticists work with both mechanical engineers and physiologists in designing new robots.

Computer science also has indirect relationships with virtually all disciplines that use computers. Applications developed in other fields often involve collaboration with computer scientists, who contribute their knowledge of algorithms, data structures, software engineering, and existing technology. In return, the computer scientists have the opportunity to observe novel applications of computers, from which they gain a deeper insight into their use. These relationships make computer science a highly interdisciplinary field of study.

3. Identify the words in bold italics:

1. A key technique ***developed*** in the study of artificial intelligence is to specify a problem as a set of states, some of which are solutions, and then search for solution states.
2. If a computer searched the states ***resulting*** from all possible sequences of moves, it could identify those that win the game.
3. However, the number of states ***associated with*** many problems (such as the possible number of moves needed to win a chess game) is so vast that exhaustively searching them is impractical.
4. Another area of computer science that has found wide practical use is *robotics*-the design and development of computer ***controlled*** mechanical devices.
5. Robots range in complexity from toys to ***automated*** factory assembly lines.
6. Roboticists-scientists ***involved*** in the field of robotics-study the many aspects of ***controlling*** robots.
7. However, a ***unifying*** theme is the development of better interfaces and experimental evaluation of their effectiveness.
8. Applications ***developed*** in other fields often involve collaboration with computer scientists.

Text 5. Basic features of database programs

1. Learn the vocabulary of the text:

1. store, organize and retrieve information
2. features and applications of a computer database
3. to enter on a database via fields
4. holds a separate piece of information
5. a record about an employee
6. length of employment
7. hold large amounts of information
8. to find records containing particular information
9. advantages of a database program
10. networking facilities
11. to have direct access to a common database
12. security devices
13. be protected by user-defined passwords
14. to import and export data

2. Read and translate the text:

With a database you can store, organize and retrieve a large collection of related information on computer. If you like, it is the electronic equivalent of an indexed filing cabinet. Let us look at some features and applications of a computer database:

• Information is entered on a database via fields. Each field holds a separate piece of information, and the fields are collected together into records. For example, a record about an employee might consist of several fields, which give his/her name, address, telephone number, age, salary, and length of employment with the company. Records are grouped together into files, which hold large amounts of information. Files can easily be updated: you can always change fields, add new records or delete old ones. With the right database software, you are able to keep track of stock, sales, market trends, orders, invoices and many more details that can make your company successful.

• Another feature of database programs is that you can automatically look up and find records containing particular information. You can also search on more than one field at a time. For example, if a managing director wanted to know all the customers that spend more than £7,000 per month, the program would search on the name field and the money field simultaneously. If we had to summarize the most relevant advantages of a database program over a card index system, we would say that it is much faster to consult and update, occupies a lot less space, and records can be automatically sorted into numerical or alphabetical order using any field. The best packages also include networking facilities, which add a new dimension of productivity to businesses. For example, managers of different departments can have direct access to a common database, which represents an enormous advantage.

Thanks to security devices, you can share part of your files on a network and control who sees the information. Most aspects of the program can be protected by user-defined passwords. For example, if you wanted to share an employee's personal details, but not his commission, you could protect the commission field. Other features like mail merging, layout design and the ability to import and export data are also very useful. In short, a database manager helps you control the data you have at home, in the library or in your business.

3. Open the brackets:

1. Information (enter) on a database via fields.
2. The fields (collect) together into records.
3. Records (group) together into files, which hold large amounts of information.
4. Files (can update): you can always change fields, add new records or delete old ones.
5. Records (can sort) into numerical or alphabetical order using any field.
6. Most aspects of the program (can protect) by user-defined passwords.

7. Most aspects of the program (can protect) by user-defined passwords.
8. Managers of different departments (can have) direct access to a common database, which represents an enormous advantage.
9. A database manager (help) you control the data you have at home, in the library or in your business.

Text 6. Computer graphics

1. Learn the vocabulary of the text:

1. to interpret the input provided by the user
2. to transform something into images
3. to convert the bits of data into precise shapes and colours
4. to use sophisticated programs
5. computer-aided design and computer-aided manufacturing
6. CAD software
7. to develop, model and test car designs
8. to save time and money
9. to present data in a more understandable form
10. to design circuits
11. present information visually
12. effective ways of communicating
13. three-dimensional graphics
14. to present information in a clear visual form.

2. Read and translate the text:

Computer graphics are pictures and drawings produced by computer. A graphics program interprets the input provided by the user and transforms it into images that can be displayed on the screen, printed on paper or transferred to microfilm. In the process the computer uses hundreds of mathematical formulas to convert the bits of data into precise shapes and colours. Graphics can be developed for a variety of uses including presentations, desktop publishing, illustrations, architectural designs and detailed engineering drawings. Mechanical engineers use sophisticated programs for applications in computer-aided design and computer-aided manufacturing. Let us take, for example, the car industry. CAD software is used to develop, model and test car designs before the actual parts are made. This can save a lot of time and money. Computers are also used to present data in a more understandable form: electrical engineers use computer graphics to design circuits and people in business can present information visually to clients in graphs and diagrams. These are much more effective ways of communicating than lists of figures or long explanations. Today, three-dimensional graphics, along with colour and animation are essential for such applications as fine art, graphic design, computer-aided engineering and academic research. Computer animation is the process of creating objects and pictures which move across the screen; it is used by scientists and engineers to analyze problems. With the appropriate software they can study the structure of objects and how it is affected by particular changes. Basically, computer graphics help users to understand complex information quickly by presenting it in a clear visual form.

3. Answer the questions:

1. What are the key words of the text?
2. What is the main idea of the text?
3. What is computer animation?
4. What is CAD?
5. How can you define "computer graphics"?
6. How is CAD used in car industry?
7. What uses can graphics be developed for?

Text 7. Programming languages

1. Learn the vocabulary of the text:

1. human-based languages
2. an array of computer programming languages
3. compiling
4. a binary
5. distinct features
6. commonalities
7. process large and complex swaths of information
8. a list of randomized numbers
9. to place in ascending order
10. the most important, relevant and in-demand languages
11. back end developers
12. flexible and robust semantics
13. frameworks
14. micro-frameworks
15. advanced content management systems
16. desktop graphical user interfaces
17. a scripting or glue language
18. to develop web-based applications
19. to develop enterprise-level applications

2. Read and translate the text:

Computer programming languages allow us to give instructions to a computer in a language the computer understands. Just as many human-based languages exist, there are an array of computer programming languages that programmers can use to communicate with a computer. The portion of the language that a computer can understand is called a "binary." Translating programming language into binary is known as "compiling." Each language, from C Language to Python, has its own distinct features, though many times there are commonalities between programming languages.

These languages allow computers to quickly and efficiently process large and complex swaths of information. For example, if a person is given a list of randomized numbers ranging from one to ten thousand and is asked to place them in ascending order, chances are that it will take a sizable amount of time and include some errors.

There are dozens of programming languages used in the industry today. We've compiled overviews of the most important, relevant and in-demand of these languages below.

PYTHON

Python is an advanced programming language that is interpreted, object-oriented and built on flexible and robust semantics.

Who uses it?

- **Professions and Industries:**
  - Python developers, software engineers, back end developers, Python programmers
  - Used by employers in information technology, engineering, professional services and design
- **Major Organizations:** Google, Pinterest, Instagram, YouTube, DropBox, NASA, ESRI
- **Specializations and Industries:** Web and Internet development (frameworks, micro-frameworks and advanced content management systems); scientific and numeric computing; desktop graphical user interfaces (GUIs)

What makes learning it important?

Python lets you work quickly to integrate systems as a scripting or glue language. It's also suited for Rapid Application Develop (RAD).

- The game Civilization 4 has all its inner logic, including AI, implemented in Python.
- NASA uses Python in its Integrated Planning System as a standard scripting language.

Features:

- Simple to learn and easily read
- Associated web frameworks for developing web-based applications
- Free interpreter and standard library available in source or binary on major platforms

Where did it start?

Python was developed in the late 1980s at CWI in the Netherlands and first released to the public in 1991.

JAVA

Java is a general-purpose, object-oriented, high-level programming language with several features that make it ideal for web-based development.

Who uses it?

- **Professions and Industries:**
  - Software engineers, Java developers
  - Used by employers in communications, education, finance, health sciences, hospitality, retail and utilities
- **Major Organizations:** V2COM, Eclipse Information Technologies, eBay, Eurotech
- **Specializations and Industries:** Internet of Things (IoT), Enterprise Architecture, Cloud Computing

What makes learning it important?

Java is used to develop enterprise-level applications for video games and mobile apps, as well as to create web-based applications with JSP (Java Server Pages). When used online, Java allows applets to be downloaded and used through a browser, which can then perform a function not normally available.

- Programs that use or are written in Java include Adobe Creative Suite, Eclipse, Lotus Notes, Minecraft and OpenOffice.
- Java is the core foundation for developing Android apps.

Features:

- Application portability
- Robust and interpreted language
- Extensive network library

Where did it start?

Originally known as Oak, Java was developed in 1990 at Sun Microsystems to add capabilities to the C++ language. Java was developed according to the principle of WORA (Write Once Run Anywhere). The language was introduced to the public in 1995 and is now owned by Oracle.

JAVASCRIPT

JavaScript is a client-side programming language that runs inside a client browser and processes commands on a computer rather than a server. It is commonly placed into an HTML or ASP file. Despite its name, JavaScript is not related to Java.

Who uses it?

- **Professions and Industries:**
  - JavaScript developers, Web developers, software engineers
  - Used by employers in Information Technology, Engineering, Design, Marketing, Finance and Healthcare
- **Major Organizations:** WordPress, Soundcloud, Khan Academy, Linkedin, Groupon, Yahoo and many others
- **Specializations and Industries Where JavaScript is Used Most:** Front End Website Development, Gaming Development

What makes learning it important?

JavaScript is used primarily in Web development to manipulate various page elements and make them more dynamic, including scrolling abilities, printing the time and date, creating a calendar and other tasks not possible through plain HTML. It can also be used to create games and APIs.

- The agency Cyber-Duck in Britain uses public APIs, created with JavaScript, to pull in data about crime and enables users to review a local area.
- Tweetmap, created by Pete Smart and Rob Hawkes using JavaScript, represents a world map that is proportionally sized according to the number of tweets.

Features:

- Basic features are easy to learn
- Multiple frameworks
- Users can reference JQuery, a comprehensive Javascript library

Where did it start?

JavaScript was designed by Netscape and originally known as LiveScript, before becoming JavaScript in 1995.

C++

C++ is a general purpose, object-oriented, middle-level programming language and is an extension of C language, which makes it possible to code C++ in a "C style". In some situations, coding can be done in either format, making C++ an example of a hybrid language.

Who uses it?

- **Professions and Industries:**
  - C++ software engineers, C++ software developers, embedded engineers, programmer analysts

- Used by employers in Information Technology, Engineering, Professional Services, Design, Quality Control and Management
- **Major Company and Organization Users:** Google, Mozilla, Firefox, Winamp, Adobe Software, Amazon, Lockheed Martin
- **Specializations:** System/Application Software, Drivers, Client-Server Applications, Embedded Firmware

What makes learning it important?

The C++ language is used to create computer programs and packaged software, such as games, office applications, graphics and video editors and operating systems.

- The Blackberry OS is developed using C++.
- The newest Microsoft Office suite was developed using C++.

Features:

- Often the first programming language taught at college level
- Quick processing and compilation mechanism
- Robust standard library (STL)

Where did it start?

Released in 1983 and often considered an object-oriented version of C language, C++ was created to compile lean, efficient code, while providing high-level abstractions to better manage large development projects.

3. Translate sentences from Russian into English:

Классификация языков программирования

1. На данный момент существует более 300 языков программирования.
2. Каждый из них имеет свои особенности и подходит для одной определенной задачи.
3. Все языки программирования можно условно разделить на несколько групп:
4. Аспектно-ориентированные (основная идея – разделение функциональности для увеличения эффективности программных модулей).
5. Структурные (в основе лежит идея создания иерархической структуры отдельных блоков программы).
6. Логические (в основе лежит теория аппарата математической логики и правил резолюции).
7. Объектно-ориентированные (в таком программировании используются уже не алгоритмы, а объекты, которые принадлежат определенному классу).
8. Мультипарадигмальные (сочетают в себе несколько парадигм, и программист сам решает, каким языком воспользоваться в том или ином случае).

9. Функциональные (в качестве основных элементов выступают функции, которые меняют значение в зависимости от результатов вычислений исходных данных).

Text 8. Debugging a computer program

1. Learn the vocabulary of the text:

1. a bug
2. an error in a software program
3. to quit or behave in an unintended manner
4. a button
5. a program's interface
6. to respond
7. to hang or crash
8. an infinite calculation
9. memory leak.
10. syntax or logic errors
11. the source code of a program
12. to fix
13. a development tool
14. a debugger
15. to negatively affect the usability of a program
16. to go through a lot of testing
17. to release
18. commercial software
19. completely error-free program
20. bug fixes for errors
21. to get rid of all the bugs
22. eliminate errors
23. a windshield at a gas station
24. debuggers
25. to mark the exact lines of code
26. to run a program
27. determine
28. provide detailed information
29. execution

2. Read and translate the text.

In the computer world, **a bug** is an error in a software program. It may cause a program to unexpectedly quit or behave in an unintended manner. For example, a small bug may cause a button within a program's interface not to respond when you click it. A more serious bug may cause the program to hang or crash due to an infinite calculation or memory leak.

From a developer perspective, bugs can be syntax or logic errors within the source code of a program. These errors can often be fixed using a development tool aptly named a debugger. However, if errors are not caught before the program is compiled into the final application, the bugs will be noticed by the user.

Because bugs can negatively affect the usability of a program, most programs typically go through a lot of testing before they are released to the public. For example, commercial software often goes through a beta phase, where multiple users thoroughly test all aspects of the program to make sure it functions correctly. Once the program is determined to be stable and free from errors, it is released the public.

Of course, as we all know, most programs are not completely error-free, even after they have been thoroughly tested. For this reason, software developers often release "point updates," (e.g. version 1.0.1), which include bug fixes for errors that were found after the software was released. Programs that are especially "buggy" may require multiple point updates (1.0.2, 1.0.3, etc.) to get rid of all the bugs.

Computer programmers, like everybody else, are not perfect. This means the programs they write sometimes have small errors, called "bugs," in them. These bugs can be minor, such as not recognizing user input, or more serious, such as a memory leak that crashes the program. Before releasing their software to the public, programmers **"debug"** their programs, eliminating as many errors as possible. This debugging process often takes a long time, as fixing some errors may introduce others. Debugging your windshield at a gas station is much easier than debugging a computer program.

Even the most experienced software programmers usually don't get it right on their first try. Certain errors, often called bugs, can occur in programs, causing them to not function as the programmer expected. Sometimes these errors are easy to fix, while some bugs are very difficult to trace. This is especially true for large programs that consist of several thousand lines of code.

Fortunately, there are programs called **debuggers** that help software developers find and eliminate bugs while they are writing programs. A debugger tells the programmer what types of errors it finds and often marks the exact lines of code where the bugs are found. Debuggers also allow programmers to run a program step by step so that they can determine exactly when and why a program crashes. Advanced debuggers provide detailed information about threads and memory being used by the program during each step of execution.

3. Translate the sentences with the word <u>Bug</u>

1. It's... a *bug*!
2. The *bug* looked to be about two feet long.
3. 'He said it looked like a bed *bug*, monsieur, but not so that the mechanics could hear what he said.
4. Wasn't no bigger'n a *bug* first time he gave me C chord.
5. "Have you sent anyone in to sweep for the *bug*?"
6. "But not our space *bug*."
7. And this *bug* mutated as a direct result of our mass consumption of animals, particularly pork.
8. Okay, *bug* report taken.
9. You mean the *bug*.
10. It's only a nasty *bug*.
11. And if we hit a rough spot, instead of getting mad, just say, "Hey, we found a *bug*," and we report it so it can be fixed.
12. I'll go get *Bug* ready for school.
13. We got a *bug* in the electrical system!
14. We're looking for a *bug*, not a password.
15. You have an "emotional *bug*".
16. Sorry, Mr. Gross, but all programs have *bugs*.
17. We might be getting some interference from the *bug*s.
18. They been chasing *bugs* ever since they installed it.
19. It was a trial run an early experiment to work out the *bugs*, so to speak.
20. It's a complex program, and there's still a *bug* or two to be worked out.

Text 9. Software testing

1. Learn the vocabulary of the text:

1. to find and fix bugs
2. to suit to a career in software testing
3. a software tester
4. to be involved in
5. software development and deployment
6. to conduct automated and manual tests
7. to ensure the software is fit for purpose

8. the analysis of software and systems
9. to avert risk and
10. to prevent software issues
11. to find bugs and issues within a product
12. to work on bespoke
13. to be familiar with programming and using coding languages
14. to assess a code
15. project requirements
16. to assess potential risks
17. a key requirement
18. work more flexibly
19. keep up to date

2. Read and translate the text:

If you love finding and fixing bugs in programming and coding, you could be suited to a career in software testing.

As a software tester, you are involved in the quality assurance stage of software development and deployment. You'll conduct automated and manual tests to ensure the software created by developers is fit for purpose. Software testing involves the analysis of software, and systems, to avert risk and prevent software issues.

Your role is integral to the creation of software systems and technical products including vehicles, electronic goods, defense, and healthcare.

Ultimately software testers are employed to find bugs and issues within a product before it gets deployed to everyday users. You might work on bespoke, individual projects or multinational projects spanning the globe and costing billions of pounds. You will need to be, or become, familiar with programming and using coding languages. Assessing code is one part of the role of a software tester.

Responsibilities

Your role will vary depending on project requirements. You may join a project at the initial implementation stages to assess potential risks, or be brought on to a project midway through, when testing becomes a key requirement.

Large organisations may have software testers dedicated to one project; whereas smaller organisations may have a central team working on multiple projects.

However, your work activities are likely to include:

- meeting with system users to understand the scope of projects
- working with software developers and project support teams
- identifying business requirements
- project planning
- monitoring applications and software systems

- stress testing
- performance testing
- functional testing
- scalability testing
- writing and executing test scripts
- running manual and automated tests
- testing in different environments including web and mobile
- writing bug reports
- resource planning
- reviewing documentation
- working towards departmental and project deadlines
- quality assurance
- providing objective feedback to software development project teams
- problem solving
- designing tests to mitigate risk
- presenting findings to software development and business user teams
- travelling to different project sites
- working on multiple projects at one time
- document analysis
- liaising with project teams in other parts of the world
- communicating findings to technical and non-technical colleagues.

Working hours

Working hours usually follow a standard office day of eight or nine hours, between 8am and 6pm. However, due to the nature of project work you may be required to work outside these times. On occasion this may mean working shifts and weekend work. This would be most likely to occur during periods of software deployment or if a project happens to be taking place across a variety of locations and time zones.

What to expect

- Work is mainly office based and you will spend the majority of your time at a computer.
- Your role may be stressful at times, particularly around the time of project completion.
- Once you have gained adequate experience, you could progress into the freelance and contracting market. This would enable you to select specific projects and work more flexibly. However, working as a contractor may not provide the same benefits and job security in comparison to a permanent employee.
- The IT sector, including software testing roles, has a higher ratio of male to female workers. However, there is a higher ratio of female to male software testers when compared with other IT jobs (such as software development).

- Companies employ software testers in many locations within the UK. The highest concentration is in large cities including London, Manchester, Edinburgh and Birmingham. There are also international opportunities, most notably in the USA and India, where a large number of off-shore software testing companies are based.

Qualifications

Software testers often have a degree in computer science or IT. However, the role is open to graduates from a variety of degree disciplines including:

- chemistry
- electrical engineering
- mathematics
- physics
- software, IT, or engineering diploma may be most highly regarded by companies.

Skills

You will need to have:

- strong verbal and written communication skills with the ability to liaise with a variety of stakeholders
- problem solving skills
- the ability to work under pressure
- attention to detail
- competent technical skills
- the ability to work in a team and individually
- organisational skills with the capability of working towards tight deadlines
- a passion for technology.

Employers

Software testers are required in a variety of organisations and sectors. Large employers with sophisticated software and IT systems will have the most opportunities. Technology companies and smaller organisations also require software testers.

You can find software testing opportunities in:

- financial services
- healthcare
- manufacturing
- media
- professional services
- public sector
- retail
- telecommunications
- transport.

Professional development

As the IT sector is ever changing, it is important that you keep up to date with developments and specific software testing trends. On-the-job training is an ideal way for students and recent graduates to gain an understanding of the software development lifecycle.

3. Translate the passage from Russian into English:

Необходимыми качествами тестировщика являются логическое мышление, внимательность, хорошая память, умение учиться и адаптироваться к существующим задачам, быстро переключаться с одного типа задач на другой. Не менее важны терпение, усидчивость и умение работать в команде.

Кроме того, тестировщик выступает одновременно и как пользователь, и как эксперт, а потому должен иметь определенный склад мышления: уметь воспроизводить поведение пользователя продукта и анализировать поведение системы, входящие параметры и полученные результаты с точки зрения инженера.

Некоторые утверждают, что специфика профессии заключается в видимом однообразии и монотонности трудового процесса; по мнению других, тестирование является творческой исследовательской работой (в противовес стандартизированной разработке).

Одной из особенностей профессии является возможность удаленной работы, причем расстояние часто не имеет значения (тестировщик может находиться в другом городе или стране по отношению к разработчику и заказчику).

Text 10. Computer crimes

1. Learn the vocabulary of the text:

1. the study of computer abuse
2. the proverbial tip of the iceberg
3. the pickings
4. dishonest employees
5. a thief
6. to gain access to funds
7. cash-dispensing terminals
8. counterfeit credit cards
9. blackmail
10. unscrupulous competitors
11. crooked computer experts
12. to devise a variety of tricks

13. unauthorized ways
14. use account numbers and passwords
15. embezzlers

2. Read and translate the text:

More and more, the operations of our businesses, governments, and financial institutions are controlled by information that exists only inside computer memories. Anyone clever enough to modify this information for his own purposes can reap substantial rewards. Even worse, a number of people who have done this and been caught at it have managed to get away without punishment.

These facts have not been lost on criminals or would-be criminals. A recent Stanford Research Institute study of computer abuse was based on 160 case histories, which probably are just the proverbial tip of the iceberg. After all, we only know about the unsuccessful crimes. How many successful ones have gone undetected is anybody's guess.

Here are a few areas in which computer criminals have found the pickings all too easy.

Banking. All but the smallest banks now keep their accounts on computer files. Someone who knows how to change the numbers in the files can transfer funds at will. For instance, one programmer was caught having the computer transfer funds from other people's accounts to his wife's checking account. Often, traditionally trained auditors don't know enough about the workings of computers to catch what is taking place right under their noses.

Business. A company that uses computers extensively offers many opportunities to both dishonest employees and clever outsiders. For instance, a thief can have the computer ship the company's products to addresses of his own choosing. Or he can have it issue checks to him or his confederates for imaginary supplies or services. People have been caught doing both.

Credit Cards. There is a trend toward using cards similar to credit cards to gain access to funds through cash-dispensing terminals. Yet, in the past, organized crime has used stolen or counterfeit credit cards to finance its operations. Banks that offer after-hours or remote banking through cash-dispensing terminals may find themselves unwillingly subsidizing organized crime.

Theft of Information. Much personal information about individuals is now stored in computer files. An unauthorized person with access to this information could use it for blackmail. Also, confidential information about a company's products or operations can be stolen and sold to unscrupulous competitors. (One attempt at the latter came to light when the competitor turned out to be scrupulous and turned in the people who were trying to sell him stolen information.)

Software Theft. The software for a computer system is often more expensive than the hardware. Yet this expensive software is all too easy to copy. Crooked computer experts have devised a variety of tricks for getting these expensive programs printed out, punched on cards, recorded on tape, or otherwise delivered into their hands. This crime has even been perpetrated from remote terminals that access the computer over the telephone.

Theft of Time-Sharing Services. When the public is given access to a system, some members of the public often discover how to use the system in unauthorized ways. For example, there are the "phone freakers" who avoid long distance telephone charges by sending over their phones control signals that are identical to those used by the telephone company. Since time-sharing systems often are accessible to anyone who dials the right telephone number, they are subject to the same kinds of manipulation.

Of course, most systems use account numbers and passwords to restrict access to authorized users. But unauthorized persons have proved to be adept at obtaining this information and using it for their own benefit. For instance, when a police computer system was demonstrated to a school class, a precocious student noted the access codes being used; later, all the student's teachers turned up on a list of wanted criminals.

Perfect Crimes. It's easy for computer crimes to go undetected if no one checks up on what the computer is doing. But even if the crime is detected, the criminal may walk away not only unpunished but with a glowing recommendation from his former employers.

Of course, we have no statistics on crimes that go undetected. But it's unsettling to note how many of the crimes we do know about were detected by accident, not by systematic audits or other security procedures. The computer criminals who have been caught may have been the victims of uncommonly bad luck.

For example, a certain keypunch operator complained of having to stay overtime to punch extra cards. Investigation revealed that the extra cards she was being asked to punch were for fraudulent transactions. In another case, disgruntled employees of the thief tipped off the company that was being robbed. An undercover narcotics agent stumbled on still another case. An employee was selling the company's merchandise on the side and using the computer to get it shipped to the buyers. While negotiating for LSD, the narcotics agent was offered a good deal on a stereo!

Unlike other embezzlers, who must leave the country, commit suicide, or go to jail, computer criminals sometimes brazen it out, demanding not only that they not be prosecuted but also that they be given good recommendations and perhaps other benefits, such as severance pay. All too often, their demands have been met.

Why? Because company executives are afraid of the bad publicity that would result if the public found out that their computer had been misused. They cringe at the thought of a criminal boasting in open court of how he juggled the most confidential records right under the noses of the company's executives, accountants, and security staff. And so, another computer criminal departs with just the recommendations he needs to continue his exploits elsewhere.

3. Identify the following verb forms:

are controlled, exists, have done, have been caught, was based, revealed, is detected, noted, have devised, can be stolen and sold, may find, have not been lost, know, was selling, could use, had been misused, departs, have been met.

Text 11. What is information security?

1. Learn the vocabulary of the text:

1. information security
2. availability
3. privacy
4. integrity of data
5. the protection of important data
6. security system
7. foolproof
8. easy access to the information
9. approved codes
10. hacking programs
11. to breach
12. to make access as secure as possible
13. a mix of upper and lowercase letters
14. gain access to secure information
15. malware, which includes computer viruses, spyware, worms
16. to steal information
17. antivirus programs
18. strong antivirus software
19. to check for any known malicious software
20. a potential virus
21. a firewall
22. to be vulnerable to attack
23. antivirus packages
24. encoding data

25. encryption systems
26. to encrypt data
27. codes and cyphers
28. legal liability

2. Read and translate the text:

Information security is the process of protecting the availability, privacy, and integrity of data. While the term often describes measures and methods of increasing computer security, it also refers to the protection of any type of important data, such as personal diaries or the classified plot details of an upcoming book. No security system is foolproof but taking basic and practical steps to protect data is critical for good information security.

Password Protection

Using passwords is one of the most basic methods of improving information security. This measure reduces the number of people who have easy access to the information, since only those with approved codes can reach it. Unfortunately, passwords are not foolproof, and hacking programs can run through millions of possible codes in just seconds. Passwords can also be breached through carelessness, such as by leaving a public computer logged into an account or using a too simple code, like "password" or "1234."

To make access as secure as possible, users should create passwords that use a mix of upper and lowercase letters, numbers, and symbols, and avoid easily guessed combinations such as birthdays or family names. People should not write down passwords on papers left near the computer and should use different passwords for each account. For better security, a computer user may want to consider switching to a new password every few months.

Antivirus and Malware Protection

One way that hackers gain access to secure information is through malware, which includes computer viruses, spyware, worms, and other programs. These pieces of code are installed on computers to steal information, limit usability, record user actions, or destroy data. Using strong antivirus software is one of the best ways of improving information security. Antivirus programs scan the system to check for any known malicious software, and most will warn the user if he or she is on a webpage that contains a potential virus. Most programs will also perform a scan of the entire system on command, identifying and destroying any harmful objects.

Most operating systems include a basic antivirus program that will help protect the computer to some degree. The most secure programs are typically those available for a monthly subscription or one-time fee, and which can be downloaded online or purchased in a store. Antivirus software can also be downloaded for free online, although these programs may offer fewer features and less protection than paid versions.

Even the best antivirus programs usually need to be updated regularly to keep up with the new malware, and most software will alert the user when a new update is available for downloading. Users must be aware of the name and contact method of each anti-virus program they own, however, as some viruses will pose as security programs in order to get an unsuspecting user to download and install more malware. Running a full computer scan on a weekly basis is a good way to weed out potentially malicious programs.

Firewalls

A firewall helps maintain computer information security by preventing unauthorized access to a network. There are several ways to do this, including by limiting the types of data allowed in and out of the network, re-routing network information through a proxy server to hide the real address of the computer, or by monitoring the characteristics of the data to determine if it's trustworthy. In essence, firewalls filter the information that passes through them, only allowing authorized content in. Specific websites, protocols (like File Transfer Protocol or FTP), and even words can be blocked from coming in, as can outside access to computers within the firewall.

Most computer operating systems include a pre-installed firewall program, but independent programs can also be purchased for additional security options. Together with an antivirus package, firewalls significantly increase information security by reducing the chance that a hacker will gain access to private data. Without a firewall, secure data is more vulnerable to attack.

Codes and Cyphers

Encoding data is one of the oldest ways of securing written information. Governments and military organizations often use encryption systems to ensure that secret messages will be unreadable if they are intercepted by the wrong person. Encryption methods can include simple substitution codes, like switching each letter for a corresponding number, or more complex systems that require complicated algorithms for decryption. As long as the code method is kept secret, encryption can be a good basic method of information security.

On computers systems, there are a number of ways to encrypt data to make it more secure. With a symmetric key system, only the sender and the receiver have the code that allows the data to be read. Public or asymmetric key encryption involves using two keys — one that is publicly available so that anyone can encrypt data with it, and one that is private, so only the person with that key can read the data that has been encoded. Secure socket layers use digital certificates, which confirm that the connected computers are who they say they are, and both symmetric and asymmetric keys to encrypt the information being passed between computers.

Legal Liability

Businesses and industries can also maintain information security by using privacy laws. Workers at a company that handles secure data may be required to sign non-disclosure agreements (NDAs), which forbid them from revealing or discussing any classified topics. If an employee attempts to give or sell secrets to a competitor or other unapproved source, the company can use the NDA as grounds for legal proceedings. The use of liability laws can help companies preserve their trademarks, internal processes, and research with some degree of reliability.

Training and Common Sense

One of the greatest dangers to computer data security is human error or ignorance. Those responsible for using or running a computer network must be carefully trained in order to avoid accidentally opening the system to hackers. In the workplace, creating a training program that includes information on existing security measures as well as permitted and prohibited computer usage can reduce breaches in internal security. Family members on a home network should be taught about running virus scans, identifying potential Internet threats, and protecting personal information online.

In business and personal behavior, the importance of maintaining information security through caution and common sense cannot be understated. A person who gives out personal information, such as a home address or telephone number, without considering the consequences may quickly find himself the victim of scams, spam, and identity theft. Likewise, a business that doesn't establish a strong chain of command for keeping data secure, or provides inadequate security training for workers, creates an unstable security system. By taking the time to ensure that data is handed out carefully and to reputable sources, the risk of a security breach can be significantly reduced.

3. Answer the questions:

1. What is critical for good information security?
2. What is information security?
3. What does the term "information security" describe?
4. What is one of the most basic methods of improving information security?
5. Who has easy access to the information?
6. Why should people not write down passwords on papers left near the computer?
7. Can antivirus software be downloaded for free online?
8. How do firewalls increase information security?

Text 12. Machine translation today and tomorrow

1. Learn the vocabulary of the text:

1. machine translation (MT)
2. the pioneer research area
3. computational linguistics
4. the automatic translation of all kinds of documents
5. apparent
6. human revision of MT output
7. the crude (unedited) MT output
8. the production of human-quality translations
9. a cost-effective option
10. to produce rough translations
11. to reduce costs
12. to improve MT output
13. reduce (or even eliminating) lexical ambiguity
14. to simplify complex sentence structures
15. enhance the comprehensibility of the original texts
16. documentation workflow
17. to make effective use of MT systems
18. to be assisted by computer-based translation support tools
19. to store and search databases
20. translator workstations
21. translation tools
22. linguistically sophisticated texts
23. unrivalled
24. to be of publishable quality
25. memoranda
26. highly specialized technical subjects

2. Read and translate the text:

The field of machine translation (MT) was the pioneer research area in computational linguistics during the 1950s and 1960s. When it began, the assumed goal was the automatic translation of all kinds of documents at a quality equaling that of the best human translators. It became apparent very soon that this goal was impossible in the foreseeable future. Human revision of MT output was essential if the results were to be published in any form. At the same time, however, it was found that for many purposes the crude (unedited) MT output could be useful to those who wanted to get a general idea of the content of a text in an unknown language as quickly as possible.

For many years, however, this latter use of MT (i.e. as a tool of assimilation, for information gathering and monitoring) was largely ignored. It was assumed that MT should be devoted only to the production of human-quality translations (for dissemination). Many large organizations have large volumes of technical and administrative documentation that have to be translated into many languages. For many years, MT with human assistance has been a cost-effective option for multinational corporations and other multilingual bodies (e.g. the European Union). MT systems produce rough translations which are then revised (post-edited) by translators. But post-editing to an acceptable quality can be expensive, and many organizations reduce costs and improve MT output by the use of 'controlled' 55 languages, i.e. by reducing (or even eliminating) lexical ambiguity and simplifying complex sentence structures – which may itself enhance the comprehensibility of the original texts. In this way, translation processes are closely linked to technical writing and integrated in the whole documentation workflow, making possible further savings in time and costs. At the same time as organizations have made effective use of MT systems, human translators have been greatly assisted by computer-based translation support tools, e.g. for terminology management, for creating in-house dictionaries and glossaries, for indexing and concordances, for post-editing facilities, and above all (since the end of the 1980s) for storing and searching databases of previously translated texts ('translation memories'). Most commonly these tools are combined in translator workstations – which often incorporate full MT systems as well. Indeed, the converse is now true: MT systems designed for large organizations are including translation memories and other translation tools. As far as systems for dissemination (publishable translations) are concerned the old distinctions between human-assisted MT and computer-aided translation are being blurred, and in the near future may be irrelevant. It is widely agreed that where translation has to be of publishable quality, both human translation and MT have their roles. Machine translation is demonstrably cost-effective for large scale and/or rapid translation of technical documentation and software localization materials. In these and many other situations, the costs of MT plus essential human preparation and revision or the costs of using computerized translation tools (workstations, translation memories, etc.) are significantly less than those of traditional human translation with no computer aids. By contrast, the human translator is (and will remain) unrivalled for non-repetitive linguistically sophisticated texts (e.g. in literature and law), and even for one-off texts in highly specialized technical subjects. However, translation does not have to be always of publishable quality. Speed and accessibility may be more important. From the beginnings of MT, unrevised translations from MT systems have been found useful for low-circulation technical reports, administrative memoranda, intelligence activities, personal correspondence, indeed whenever a document is to be read by just one or two people interested only in the essential message and unconcerned about stylistic quality or even exact terminology. The range of options has expanded significantly since the early 1990s, with the increasing use and rapid development of personal computers and the Internet.

3. Find synonyms for the following adjectives:

Equal, useful, general, quick, rough, expensive, complex, significant, powerful, real.

4. Translate the following word combinations:

the pioneer research area, human-quality translations, human assistance, cost-effective option, rough translations, complex sentence structures, documentation workflow, computer-based translation support tools, in-house dictionaries and glossaries, translator workstations, translation memories and other translation tools, non-repetitive linguistically sophisticated texts, low-circulation technical reports, rapid development.

5. Make up special questions to the given sentences:

1. The assumed goal was the automatic translation of all kinds of documents at a quality equaling that of the best human translators. (What?)
2. Human revision of MT output was essential if the results were to be published in any form. (Why?)
3. The crude (unedited) MT output could be useful to those who wanted to get a general idea of the content of a text in an unknown language as quickly as possible. (What? Who?)
4. Many large organizations have large volumes of technical and administrative documentation that have to be translated into many languages. (What?)
5. MT systems produce rough translations which are then revised (post-edited) by translators. (Who?)
6. Many organizations reduce costs and improve MT output by the use of 'controlled' 55 languages. (How?)
7. MT systems designed for large organizations are including translation memories and other translation tools. (What?)
8. The human translator is (and will remain) unrivalled for non-repetitive linguistically sophisticated texts. (Why?)
9. Speed and accessibility may be more important. (Why?)
10. The range of options has expanded significantly since the early 1990s, with the increasing use and rapid development of personal computers and the Internet. (When?)

UNIT 3. THE PECULIARITIES OF TRANSLATING MATH TEXTS (ОСОБЕННОСТИ ПЕРЕВОДА МАТЕМАТИЧЕСКИХ ТЕКСТОВ).

1. Стилистические особенности математических текстов: основные теоретические сведения

Символический язык

Тексты по математике характеризуются параллельным использованием вербальных и невербальных выражений, например: One solution is y = x 2, because its derivative is 2x. But the derivatives of x 2+3, x 2−1/2, and so on are also equal to 2x. Небески (1977) утверждает, что символический язык используется в математике там, где естественный, несимвольный язык кажется ненадежным для выражения математической идеи. Символизм используется не только для того, чтобы сократить идеи (сложная мысль, выраженная минимальным количеством символов), но и для того, чтобы указать на структуру идеи. Это легко можно увидеть в таких структурах, которые являются не линейными, а двумерными (например, символическое выражение матриц). Люди, в которых преобладает математический и логический интеллект, способны оценивать количество, легко понимать символы и символический язык, выполнять абстрактные операции.

Терминология

Математика занимает значительное место среди других наук благодаря своей строго аксиоматической структуре, дедуктивным рассуждениям и очень точной терминологии. Каждый термин в математике глубоко определен до такой степени, что никакие двусмысленности недопустимы. Эта тенденция возможна только потому, что математика в любой точке мира описывает и интерпретирует одну и ту же абстрактную реальность, независимо от внешних (социально детерминированных) влияний. Удивительно, но это явление не устраняет ряд расхождений в математической терминологии двух разных языков.

Грамматика

Математические тексты отличаются от текстов других наук использованием специфических грамматических средств. Довольно часто используются повелительные формы от первого лица единственного числа (например, Let us consider, Let’s define) или формы будущего времени от первого лица множественного числа (например, Мы рассмотрим, Мы выберем). Они обычно используются при указании производительности. В математическом тексте принято избегать использования более традиционной перформативной формы, то есть указательной формы настоящего времени от первого лица единственного или множественного числа.

Синтаксис

Одной из главных особенностей математических текстов является то, что обычный текст, как правило, перемежается большим количеством малых и больших громоздких формул, иногда занимающих целые страницы, что делает практически невозможными многие филологические исследования математических текстов. Наличие формул приводит к особенностям синтаксиса, характерным только для научного языка математики. К ним относится свободная позиция формул на месте главных и второстепенных членов предложения и связанная с этим специфика сильного обособления. Ради краткости, термины и словосочетания, сопровождающие формулы и номера формул, стали часто опускаться на письме в тех случаях, когда они легко восстанавливаются из контекста. Эта особенность научного языка математики приводит к неоднозначности устной реализации математического текста и к особенностям его синтаксиса.

Низкий эмоциональный заряд

Наряду с сокрытием личности автора, математические тексты отличаются низким эмоциональным зарядом. Автор избегает показывать какое-либо личное отношение к теме ради объективности. Как утверждает Чехова (2008), в научных текстах используется чисто стандартный код, поскольку он не позволяет авторам выражать идеи эмоциональным тоном.

Состав

Абзацы в математических текстах очень связны и касаются только конкретной идеи. Все предложения логически объединены (не обязательно с помощью соединителей), чтобы вызвать или вывести новые идеи.

Текстовая графика

Из-за сложности математических текстов для того, чтобы сделать их понятными, используются графические средства (система скобок, надписи, нумерация, чрезмерное использование знаков препинания, символов и других знаков).

Языковая экономика

Авторы математических текстов уплотняют всю информацию, чтобы она была как можно более точной. Используется меньше слов, чтобы избежать избыточной информации.

2. Способы преодоления интерференции при переводе текстов по математике

Сущность явления интерференции состоит в том, что, когда человек усваивает иностранный язык, он неосознанно переносит на этот язык систему действующих правил и программу речевого поведения, существующие в родном языке. Интерференция определяется как совокупность различных признаков выражения данного смысла в двух сопоставительных системах, образующих третью, в которой действуют законы родного и неродного языков. Сложность усвоения и закрепления совокупности различных признаков этой третьей системы и вызывает интерференцию.

С методической точки зрения интерференцию можно определить, как непроизвольное допущение учащимися в речи на неродном языке различных неточностей с точки зрения нормы изучаемого языка как результат отрицательного влияния родного языка. В этом случае ранее приобретенные речевые умения и навыки не способствуют, а, наоборот, препятствуют формированию новых речевых умений и навыков на изучаемом языке.

Для преодоления *интерференции в переводе* используются переводческие трансформации. Один из видов таких трансформаций – лексические, суть которых заключается в замене переводимой лексической единицы словом или словосочетанием, которое реализует сему данной единицы исходного языка (ИЯ). В данном пособии приемы лексических трансформаций рассматриваются на примере текстов по математике, которые не использовались ранее в качестве материала для анализа интерференции в переводе. Задача переводчика сделать так, чтобы текст перевода соответствовал нормам языка перевода, и при этом сохранил исходный текст (то, ради чего был создан оригинал), и, следовательно, переводчику необходимо постараться снизить или преодолеть влияние интерференции.

Приемы логического мышления, с помощью которых переводчик раскрывает значение иностранного слова в контексте и находит ему русское соответствие, принято называть лексическими трансформациями. *Лексические трансформации* – это отклонение при переводе от словарных соответствий, которое заключается в замене отдельных лексических единиц исходного языка на лексические единицы переводного языка, не являющиеся их эквивалентами. Рассмотрим причины, по которым переводчики прибегают к использованию лексических трансформаций:

Во-первых, *в двух языках значение одного слова определяется разными признаками*.

Например, английское словосочетание ***simply connected*** в текстах по математике будет переводиться на русский как ***односвязный***, хотя в словаре у слова simply нет значения одно; англ. ***cross product*** переводится как ***векторное произведение***, хотя слово cross – поперечный, перекрестный не имеет отношения к слову vector – векторный; англ. ***value of a vector*** переводится как ***модуль вектора***; англ. ***solid geometry*** – рус. ***геометрия в пространстве***; англ. ***exact division*** – рус. ***деление без остатка***; англ. ***partial fraction*** – рус. ***элементарная дробь***. англ. ***rough classification*** – рус. ***приблизительная классификация***.

Во-вторых, смысловой объем слова в двух языках может быть не одинаков. Например, английское слово ***volume*** имеет целый ряд значений: это и ***том***, и ***громкость***, и ***объем (мат.).*** В русском языке каждому значению соответствует отдельное слово, которое будет выбираться в зависимости от контекста.

В-третьих, в двух языках сочетаемость слов может сильно отличаться. Так словосочетание ***principal root*** в математических текстах будет переводиться как ***арифметическое значение корня***, потому что выражение главный или принципиальный корень звучит нелепо; ***обобщенные функции*** на английском языке **–** ***distributions***, а не generalized functions; англ. ***rough classification*** **–** рус. ***приблизительная классификация***, а не грубая.

Нередко приемы трансформаций совмещаются, например, первые два. Все лексические трансформации основываются на формально-логических отношениях между понятиями. Приемы логических трансформаций базируются на таких формально-логических категориях как подчинение, контрадикторность, перекрещивание, внеположенность.

Под *лингвистической интерференцией* следует понимать взаимовлияние контактирующих языков, которое может быть, как отрицательным, так и положительным и выражаться в отклонениях от нормы в одном языке под влиянием другого (при отрицательной интерференции) и в приобретении, закреплении и усилении навыков в одном языке под влиянием другого (при положительной интерференции). То есть проявлением негативного влияния интерференции являются ошибки, допускаемые в переводе. Следовательно, *знание и корректное использование приемов лексических трансформаций помогает преодолеть отрицательную интерференцию и делает коммуникацию успешной.* Это характерно не только для общеупотребительной лексики, но и для узко специализированных текстов, например, по математике. Примеры, представленные ниже, были отобраны с использованием англо-русского словаря математических терминов П.С. Александрова и Г.И. Беликовой, а также электронного словаря translate. google.

Например,

точка – point (мат.), dot, full stop, spot;
угол – angle (мат.), corner;
круг – circle (мат.), round, lap; пара – pair (мат.), couple;
цифра – numeral, number, figure, digit;
острый – acute (мат. угол), sharp;
тупой – blunt, stupid, dull, obtuse (мат.);
выводить – to deduce, to derive, to infer, to conclude;
простой – simple, prime (мат. число);
скобки – braces, brackets, parentheses;
получать – to get, to obtain;
заменить – to replace, to substitute, to change;
рассматриваемое – regarded, considered, viewed;
луч – beam, ray (мат.);

решать – to decide, to solve (мат.);
слагаемое – summand, addend;
количество – number, quantity;
снабженное – equipped, supplied.
И, наоборот, *семантически недифференцированными* оказываются английские слова:
plane – самолет, плоскость (мат.);
to vanish – исчезать, стремиться к нулю (мат.);
statement – заявление, утверждение (мат.);
letter – письмо, буква;
degree – степень, градус (мат.);
square – квадрат, площадь;
kite – воздушный змей, гладкий ромб (мат.);
ratio – отношение, коэффициент;
locus – местоположение, геометрическое место точек (мат.);
volume – том, громкость, объем (мат.);
table – стол, таблица (мат.);
ruler – правитель, линейка (мат.);
compass – компас, циркуль;
edge – край, преимущество, ребро (мат.);
even – ровный, четный (мат.);
power – власть, мощность, степень (мат.);
proposition – утверждение, предложение, теорема (мат.).

Распространенность приемов *дифференциации и конкретизации* при переводе с английского языка на русский объясняется обилием в английском языке слов с широкой семантикой, которым нет прямого соответствия в русском языке. Конкретизация невозможна без дифференциации. Конкретизацией называется замена слов или словосочетания ИЯ с более широким референциальным значением словом или словосочетанием ПЯ с более узким значением. Конкретизация может быть языковой и контекстуальной. При языковой конкретизации замена слова с широким значением словом с более узким значением обуславливается расхождениями в строе двух языков; отсутствием в ПЯ лексической единицы, имеющей столь же широкое значение, что и передаваемая единица ИЯ; расхождением в стилистической характеристике; требованиями грамматического порядка (например, замена именного сказуемого глагольным).

Так, английское существительное ***figure*** имеет очень абстрактное значение. Оксфордский словарь английского языка определяет его как:
1) a number, especially one that forms part of official statistics or relates to the financial performance of a company;
2) a person's bodily shape, especially that of a woman and when considered to be attractive

Оно переводится на русский язык как: *рисунок, цифра, фигура, диаграмма, образ, личность, изображение, чертеж, вид, иллюстрация, форма.*

Конкретизируются при переводе на русский язык глаголы *calculate, compute*. Эти глаголы в отличие от русских глаголов не включают в свою семантику компонента, указывающего на способ вычисления, поэтому calculate при переводе конкретизируется как «вычислять на калькуляторе»; а compute – «вычислять на компьютере». Обычной является конкретизация глаголов *decide и solve*, которые могут переводиться как «решать, принимать решение, разрешать, находить выход»:

1) We haven't completely solved the question of what configurations occur. – Мы не полностью решили вопрос о том, какие конфигурации происходят.

2) He has to decide what to do next. – Он должен решить, что делать дальше. Прием конкретизации используется также и в передаче других слов с широким значением:

For example, given a real number x and a positive integer N ... – Например, дано действительное число x и положительное целое число N …. Английским *real* и *positive* соответствуют русские «реальный» и «позитивный», однако эти выражения стилистически неприемлемы в данном учебно-математическом контексте; отсюда необходимость конкретизации при переводе.

Рассмотрим конкретизацию глагола be:

One obvious way is to give the rules of inference. – Один из очевидных путей ее решения состоит в задании определенных правил вывода. (В другом контексте могло бы быть опущено «Один из очевидных путей решения – задать определенные правила вывода»).

Our justification for this choice is that it formalizes most of the logical principles … – Основанием для такого выбора служит то обстоятельство, что большая часть логических принципов получает свое формальное выражение …. ***Контекстуальная конкретизация*** обусловливается факторами данного конкретного контекста, чаще всего стилистическими соображениями:

− необходимость завершения фразы;

− стремление избежать повторений;

− стремление достичь большей образности, наглядности и др.

Таким образом, *интерференция* при переводе является неизбежным явлением. В частности, в примерах, упомянутых выше, можно говорить о лексической интерференции в переводе, которая чаще всего связана с объемом значения лексических единиц. В наибольшей степени она может влиять на процесс коммуникации в случаях несоответствия эмоционально-оценочного компонента двух единиц. Лексическая единица языка перевода может не иметь в одной из культур определенного компонента значения. Вместе с тем механизм перевода вполне способен это преодолеть. В работе опытного переводчика выбор переводческих трансформаций осуществляется интуитивно: переводчик не думает о том, какую трансформацию употребить.

3. Ложные друзья переводчика

Это название относится к ошибочным, ложным эквивалентным словам в языке оригинала и языке перевода. При переводе данной категории слов могут происходить ложные отождествления на основе некоторой общности слов разных языков (графической, фонетической, грамматической, а часто и семантической, т.е. общности значений). Наибольшее количество ошибок возникает при переводе интернациональной лексики (слов, заимствованных разными языками из одного источника). Интернациональные слова характеризуются общностью смысловой структуры и поэтому легко отождествляются при переводе. Однако наряду с общностью, в их значениях имеются и существенные различия, о которых переводчик часто забывает. Ниже приводится список наиболее распространенных случаев ложного отождествления, или иначе, слов, относящихся к категории ложных друзей переводчика. В первой колонке приведено исходное интернациональное слово; во второй – тот вариант его перевода, который автоматически приходит в голову неопытному переводчику; в третьей колонке даны варианты значений, в которых эти слова могут реализовываться в контексте. Это не означает, что данные слова вообще не имеют значений, упомянутых во второй колонке. Нередко они входят в смысловую структуру слова и реализуются в определенных условиях употребления. Просто в специальном тексте интернациональные слова очень часто используются в других значениях, отличающихся от тех, которые в первую очередь ассоциируются с привычным для русскоговорящих людей звуковым и графическим образом слова.

Примеры:

academic (академический) официальный, фундаментальный
activity (активность) работа
actual (актуальный) действительный
address (адресовать) рассматривать, посвящать
adequate (адекватный) правильный, надлежащий, приемлемый
alternatively (альтернативно) с другой стороны, также, кроме того
ambitious (амбициозный) перспективный
analytically (аналитически) теоретически
arena (арена) область, сфера
argue (аргументировать) считать
attribute (атрибут) характеристика
avenue (авеню) способ, направление
balance (баланс) соотношение, сочетание
basically (на базе) в основном; по существу
bonus (бонус) достоинство, преимущество
career (карьера) профессия; работа
classical (классический) традиционный

collective (коллективный) совокупный
combination (комбинация) сочетание
combined (комбинированный) общий
compromise (компромисс) соглашение сторон
concentrate (концентрировать) изучать; уделять основное внимание
concept (концепция) понятие, идея
conceptually (концептуально) согласно принятому представлению
conservative (консервативный) традиционный
context (контекст) ситуация; среда, окружение
contribution (контрибуция) достижение, вклад
conventionally (конвенционально) принято считать
conversely (противоположно) напротив, с другой стороны
critical (критический) существенный, значительный
detailed (детальный) подробное, глубокое (изучение)
dictate (диктовать) предписывать
dispute (диспут) правовой спор
dramatically (драматично) значительно, сильно
global (глобальный) всеобщий
historically (исторически) традиционно, всегда
history (история) прошлое
horizon (горизонт) перспектива, возможность
idea (идея) представление
ideally (идеально) теоретически
inadequate (неадекватный) неправильный, непригодный
informative (информационный) содержательный
instructive (инструктивный) полезный, информативный
intelligent (интеллигентный) умный
international (интернациональный) международный
intimate (интимный) глубокий (о знаниях)
intimately (интимно) тесно, непосредственно
intrigue (интриговать) вызывать интерес, привлекать внимание
intriguing (интригующий) интересный, вызывающий интерес
intuitively (интуитивно) на первый взгляд, вполне естественно
ironic (иронический) необычный
limitation (лимитирование) недостаток
marginally (маргинально) немного, минимально
massive (массированный) сильный
materialize (материализовать (ся)) становиться фактом
modality (модальность) задача, метод, вид, область, возможность, особенность
normally (нормально) обычно, как правило
organize (организовывать) составлять, состоять, связывать, разделять, разбивать, упорядочивать
originally (оригинально) первоначально
progress (прогресс) успехи

revolutionary (революционный) существенный, радикальный; имеющий важное значение
routine (рутина) процедура
specifically (специфически) в частности
technical (технический) специальный
technically (технически) формально
theoretically (теоретически) в идеале
traditionally (традиционно) обычно
trivially (тривиально) легко
unique (уникальный) особенный, исключительный

4. Перевод имён собственных (Proper nouns)

В связи с тем, что английский язык использует латинский алфавит, а русский – кириллицу, при письменном переводе с английского языка возникают трудности, связанные с передачей имен собственных и названий. Наверняка, все обращали внимание на то, что одно и то же английское имя собственное по-разному передается на русский язык. Например, Вильям и Уильям (William), Гексли и Хаксли (Huxley), Ватсон и Уотсон (Watson).

Вот несколько примеров имён известных математиков:
Nicolaus Copernicus [ˈnɪkələs kə'pɜːnɪkəs] – Николай Коперник
Johannes Kepler [jə'hænɪs ˈkeplə] – Иоганн Кеплер
Galileo [ˌgælɪ'leɪəʊ] – Галилей
Isaac Newton ['aɪzək 'nju:t(ə)n] – Исаак Ньютон
John Napier [ʤɔn 'neɪpiə] – Джон Напье
Justus Byrgius ['ʤʌstəs 'bɜːʤəs] – Юстас Бирджес
Gottfried Wilhelm Leibniz [ˈɡɒtfrid ˈwɪlˌhelm ˈliːbˌnɪz] – Го́тфрид Ви́льгельм Ле́йбниц
Isaac Barrow [ˈaɪzək ˈbærəʊ] – Исаак Барроу
Rene Descartes [rəˈneɪ deˈkɑːt] Рене Декарт
Pierre de Fermat [ˈpɪr ˈdeɪ fɛ:(r) ˈmɑː] – Пьер де Ферма
Joseph Louis Lagrange [ˈʤəʊzɪf ˈluɪs ˈlægreɪnʤ] – Жозеф Луи Лагранж
Leonard Euler [ˈlenəd ˈɔɪlər] – Леонард Эйлер
Carl Frederich Gauss [kɑːl fredˈrɪk gaʊs] – Карл Фридрих Гаусс
Augustin Louis Cauchy [ɔːgʌstɪn ˈluɪs ˈkɔːkɪ] – Августин Луи Коши Karl Weierstrass ['kɑːl wɪˈstreɪs] – Карл Вейерштрасс
Jean Baptiste Fourier [ʤiːn bəptɪst ˈfjʊərɪeɪ] – Жан Баптист Фурье
Georg Cantor [ˈgeɪəg ˈkæntər] – Георг Кантор
Julius Dedekind [ˈʤuːlɪəs ˌdediːˈkɪnd] – Юлиус Дедекинд

5. Перевод наиболее употребительных словосочетаний с предлогами и союзами в математическом английском

English	Russian
at the point x	в точке x
a group of transformations	группа преобразований
a solution of (to) equation (2.1)	решение уравнения (2.1)
at the distance of h from X	на расстоянии h от X
by assumption	по условию
by definition	по определению
by the theorem	по теореме
change x to y	заменить x на y
coincides with	совпадает с
consider the sum from l to n	рассмотрим сумму от l до n
expressed as	выраженный в виде
extend f to X	продолжить f на X
follows from	следует из
function of x	функция переменной x
f is a map from x to y	f – отображение x в y
H is determined (defined) by x	H определяется пространством x
identified with	тождественный с
integrate from a to b	интегрируем от a до b
in the case	в случае
l is parallel to n	l параллельна n
neighbourhood of x	окрестность точки x
one of the sets	одно из множеств
regarded (viewed) as a function	рассматриваемая как функция
replace x by y	заменить x на y
represent in the form	представить в виде
restrict f to A	ограничить f на A
results from the paper	результаты из статьи
substitute y for x	заменить x на y
system of equations	система уравнений
the application of x to the situation	применение x к этой ситуации
the center of the circle	центр окружности
the contribution of K to ...	вклад K в ...
the inverse for f	обратное к f
the problem for H	задача для H
the sum of a and b	сумма a и b
under the condition	при условии
x belongs to the subgroup H	x принадлежит подгруппе H
x is contained in Z	x содержится в Z
x corresponds to y	x соответствует y
x maps to y	x отображается в y
x tends to 0	x стремится к 0

6. Предлоги, союзы и обороты, характерные для научной речи

В текстах по специальности нередко используются слова, гораздо реже встречающиеся за пределами научной литературы. Некоторые из них являются архаичными формами, и об их значении, порой, трудно догадаться, опираясь на составляющие их элементы, даже если по отдельности они представляют собой знакомые лексические единицы. Такие слова часто характеризуются многозначностью, и помочь их правильно перевести позволяет контекст и знание их основных, наиболее часто реализующихся значений.

Consider – рассмотрим
due to – по причине, в связи с
furthermore – кроме того, к тому же, более того, при этом
hence – отсюда, поэтому
henceforth – далее, в дальнейшем, в последующем
in effect – в действительности, в сущности, по существу, по сути, на самом деле, фактически
in spite of, despite – несмотря на
in the sense that – в том смысле, что
on account of – вследствие, в результате, благодаря
the former ... the latter – первый ... последний
thereby – таким образом, тем самым, при этом
therefore – оттого, поэтому, отсюда, ввиду этого
to be subject to – подвергаться, подлежать
to subject – подвергать
to suppose that – допустим, что
thus – таким– образом, так что
whereas – тогда как, между тем как, в то время как, принимая во внимание, поскольку
with respect to – что касается, в отношении, в части, в сфере, применительно к

7. Методические указания студентам по чтению и переводу профессионально-ориентированных текстов

1. Перед тем, как начать переводить текст, прочтите его, начиная с заглавия, и постарайтесь понять, как можно больше без помощи словаря. Когда вы поймете, о чем идет речь, вам легче будет подобрать нужное слово или найти его в словаре.

2. Проанализируйте синтаксическую структуру предложения, выделив, в первую очередь, главные члены предложения.

3. Помните, что в словаре может не оказаться точного значения слова, необходимого вам для перевода предложения на русский язык. Словарь подскажет, в каком направлении следует вести поиск необходимого эквивалента английского слова в русском языке.

4. Не переводите пословно, т.е. слово в слово, поскольку объемы значений слов не совпадают в английском и русском языках и это может привести к искажению смысла.

5. Не следует стремиться к тому, чтобы количество слов в тексте перевода полностью соответствовало количеству слов в тексте оригинала; существительное не обязательно должно соответствовать существительному, глагол глаголу и т.д.

6. Полезно всегда помнить о том, что мы переводим не слова, а смыслы.

7. Помните, что при переводе текста всегда нужно что-то добавить, что-то поменять местами, что-то заменить, а что-то просто оставить непереведенным.

8. Следует избегать двух крайностей: как дословного, так и слишком вольного перевода. Результатом первого может быть нарушение стилистических норм родного языка. Второе может привести к частичному несоответствию текста оригинала и перевода, а, возможно, и к существенному отступлению от основного смысла исходного текста.

9. Предлоги, союзы не следует недооценивать. В английском языке они играют особо важную роль, восполняя отсутствие окончаний и согласования, важных для понимания текста.

10. Нередко слово кажется знакомым, но лишь в одном из его значений. В затруднительных случаях, если все слова знакомы, а смысл не складывается, следует проверить по словарю, нет ли у переводимых слов каких-либо других значений.

11. Нельзя переводить английское предложение так, как будто оно существует отдельно и не связано с другими предложениями в тексте. Каждое предложение всегда связано с предыдущим и воспринимается в рамках предшествующего контекста. Критерием адекватности перевода может быть положительный ответ на заданный себе вопрос: А, говорят ли так по-русски?

12. При ознакомительном чтении можно в целях экономии времени обойтись рабочим переводом текста. Если же задание предусматривает чтение и перевод вслух отрывка из текста на усмотрение преподавателя, целесообразно заранее подобрать подходящий вариант перевода, соответствующий нормам родного языка. Понимание текста само по себе ещё не свидетельствует о готовности дать литературно-обработанный вариант его перевода. Поиск адекватных русско-английских соответствий, между тем, является более сложной задачей, решение которой способствует повышению уровня владения как иностранным, так и родным языком.

13. Текст перевода всегда должен отвечать основным требованиям, которыми являются: 1) Точность. Переводчик обязан донести до читателя полностью все мысли, высказанные автором. При этом должны быть сохранены не только ос-

новные положения, но также нюансы и оттенки высказывания. 2) Лаконичность и ясность. Перевод должен быть облечен в максимально сжатую форму и изложен понятным, не затрудняющим восприятие, языком. 3) Соответствие литературным нормам родного языка. Каждая фраза должна звучать живо и естественно, не сохраняя никаких намеков на чуждые русскому языку синтаксические конструкции подлинника.

14. В целом, для успешного овладения искусством перевода необходимо соответствовать следующим условиям:

– уметь использовать знание иностранного языка

– владеть нормами родного языка

– хорошо знать дисциплину избранной специальности.

Список использованной литературы

1. Вахитова И. А. Специальные тексты по английскому языку: Для студентов физико-математического факультета: учебное издание. – Уфа: Вагант, 2005. – 84 с.

2. Глушко М. М., Выгонская Л. Н., Перекальская Т. К. Учебник английского языка для студентов-математиков старших курсов. – М.: Изд-во МГУ, 1992. – 176 с.

3. Касаткина Т. Ю. English grammar files for Math students: учеб.-метод. пособие. – Ижевск: Удмуртский университет, 2022. – 102с.

4. Касаткина Т. Ю. Mathematics and Computer Science in English : учеб.-метод. пособие. – Ижевск : Удмуртский университет, 2019. – 144с.

5. Корнилова С. А., Исмаева Ф. Х. Basic English for Mathematicians and Computer Science Learners: учебно-методическое пособие. – Казань: КФУ, 2016. – 194 с.

6. Орешина З. Д. Английский язык для студентов-математиков: учебно-методическое пособие. – Воронеж: Издательско-полиграфический центр Воронежского государственного университета, 2010. – 37 с.

7. Паникарова Н. Ф. Английский язык в информационных технологиях: учебное пособие по информатике. – Красноярск: Сибирский федеральный университет, 2007. – 133 с.

8. Потапова, Н.Л. Лексические трансформации как способ преодоления интерференции при переводе научных текстов по математике / Н.Л. Потапова // Вестник Полоцкого государственного университета. Серия А. Гуманитарные науки. – 2020. – № 2. – С. 122– 126.

9. Пушкина Е. Н. Перевод англоязычных текстов математической направленности: Учебно-методическое пособие. – Нижний Новгород: Нижегородский госуниверситет, 2021. – 168 с.

10. Сосинский А. Б. Mathematical English: Учебник английского языка для математиков. – М.: МЦНМО, 2007. – 88 с.

11. Татаринов, В.В. Краткий русско-английский словарь по элементарной математике. – Москва: Постер-М, 2022. – 36 с.

ОПИСАНИЕ ФУНКЦИОНАЛЬНОСТИ ИЗДАНИЯ:

Интерфейс электронного издания (в формате pdf) можно условно разделить на 2 части.

Левая навигационная часть (закладки) включает в себя содержание книги с возможностью перехода к тексту соответствующей главы по левому щелчку компьютерной мыши.

Центральная часть отображает содержание текущего раздела. В тексте могут использоваться ссылки, позволяющие более подробно раскрыть содержание некоторых понятий.

МИНИМАЛЬНЫЕ СИСТЕМНЫЕ ТРЕБОВАНИЯ:

Celeron 1600 Mhz; 128 Мб RAM; Windows XP/7/8 и выше; 8x DVD-ROM; разрешение экрана 1024×768 или выше; программа для просмотра pdf.

СВЕДЕНИЯ О ЛИЦАХ, ОСУЩЕСТВЛЯВШИХ ТЕХНИЧЕСКУЮ ОБРАБОТКУ И ПОДГОТОВКУ МАТЕРИАЛОВ:

Оформление электронного издания : Издательский центр «Удмуртский университет».

Авторская редакция.

Подписано к использованию 18.12.2025
Объем электронного издания Мб
Издательский центр «Удмуртский университет»
426034, г. Ижевск, ул. Ломоносова, д. 4Б, каб. 021
Тел. : +7(3412)916-364 E-mail: editorial@udsu.ru