

Минобрнауки России
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тульский государственный университет»
Интернет-институт ТулГУ
Кафедра иностранных языков

Контрольная работа № 1
по дисциплине «Иностранный язык (английский язык)»

Вариант 3

Выполнил:

Артемов Александр Евгеньевич,
гр. ИБ262521-ф,
прикладная информатика в промышленности

Проверил:

Исаева Анастасия Юрьевна,
канд. филол. наук, доц.

Тула - 2022 год

Ссылка на источник:

<https://www.britannica.com/science/computer-science>

Computer science

Computer science, the study of computers and computing, including their theoretical and algorithmic foundations, hardware and software, and their uses for processing information. The discipline of computer science includes the study of algorithms and data structures, computer and network design, modeling data and information processes, and artificial intelligence. Computer science draws some of its foundations from mathematics and engineering and therefore incorporates techniques from areas such as queueing theory, probability and statistics, and electronic circuit design. Computer science also makes heavy use of hypothesis testing and experimentation during the conceptualization, design, measurement, and refinement of new algorithms, information structures, and computer architectures.

Computer science is considered as part of a family of five separate yet interrelated disciplines: computer engineering, computer science, information systems, information technology, and software engineering. This family has come to be known collectively as the discipline of computing. These five disciplines are interrelated in the sense that computing is their object of study, but they are separate since each has its own research perspective and curricular focus. (Since 1991 the Association for Computing Machinery [ACM], the IEEE Computer Society [IEEE-CS], and the Association for Information Systems [AIS] have collaborated to develop and update the taxonomy of these five interrelated disciplines and the guidelines that educational institutions worldwide use for their undergraduate, graduate, and research programs.)

The major subfields of computer science include the traditional study of computer architecture, programming languages, and software development. However, they also include computational science (the use of algorithmic techniques for modeling scientific data), graphics and visualization, human-computer interaction, databases and information systems, networks, and the social and professional issues that are unique to the practice of computer science. As may be evident, some of these subfields overlap in their activities with other modern fields, such as bioinformatics and computational chemistry. These overlaps are the consequence of a tendency among computer scientists to recognize and act upon their field's many interdisciplinary connections.

Development of computer science

Computer science emerged as an independent discipline in the early 1960s, although the electronic digital computer that is the object of its study was invented some two decades earlier. The roots of computer science lie primarily in the related fields of mathematics, electrical engineering, physics, and management information systems.

Mathematics is the source of two key concepts in the development of the computer—the idea that all information can be represented as sequences of zeros

and ones and the abstract notion of a “stored program.” In the binary number system, numbers are represented by a sequence of the binary digits 0 and 1 in the same way that numbers in the familiar decimal system are represented using the digits 0 through 9. The relative ease with which two states (e.g., high and low voltage) can be realized in electrical and electronic devices led naturally to the binary digit, or bit, becoming the basic unit of data storage and transmission in a computer system.

Electrical engineering provides the basics of circuit design—namely, the idea that electrical impulses input to a circuit can be combined using Boolean algebra to produce arbitrary outputs. (The Boolean algebra developed in the 19th century supplied a formalism for designing a circuit with binary input values of zeros and ones [false or true, respectively, in the terminology of logic] to yield any desired combination of zeros and ones as output.) The invention of the transistor and the miniaturization of circuits, along with the invention of electronic, magnetic, and optical media for the storage and transmission of information, resulted from advances in electrical engineering and physics.

Management information systems, originally called data processing systems, provided early ideas from which various computer science concepts such as sorting, searching, databases, information retrieval, and graphical user interfaces evolved. Large corporations housed computers that stored information that was central to the activities of running a business—payroll, accounting, inventory management, production control, shipping, and receiving.

Theoretical work on computability, which began in the 1930s, provided the needed extension of these advances to the design of whole machines; a milestone was the 1936 specification of the Turing machine (a theoretical computational model that carries out instructions represented as a series of zeros and ones) by the British mathematician Alan Turing and his proof of the model’s computational power. Another breakthrough was the concept of the stored-program computer, usually credited to Hungarian American mathematician John von Neumann. These are the origins of the computer science field that later became known as architecture and organization.

In the 1950s, most computer users worked either in scientific research labs or in large corporations. The former group used computers to help them make complex mathematical calculations (e.g., missile trajectories), while the latter group used computers to manage large amounts of corporate data (e.g., payrolls and inventories). Both groups quickly learned that writing programs in the machine language of zeros and ones was not practical or reliable. This discovery led to the development of assembly language in the early 1950s, which allows programmers to use symbols for instructions (e.g., ADD for addition) and variables (e.g., X). Another program, known as an assembler, translated these symbolic programs into an equivalent binary program whose steps the computer could carry out, or “execute.”

Other system software elements known as linking loaders were developed to combine pieces of assembled code and load them into the computer’s memory, where they could be executed. The concept of linking separate pieces of code was

important, since it allowed “libraries” of programs for carrying out common tasks to be reused. This was a first step in the development of the computer science field called software engineering.

Later in the 1950s, assembly language was found to be so cumbersome that the development of high-level languages (closer to natural languages) began to support easier, faster programming. FORTRAN emerged as the main high-level language for scientific programming, while COBOL became the main language for business programming. These languages carried with them the need for different software, called compilers, that translate high-level language programs into machine code. As programming languages became more powerful and abstract, building compilers that create high-quality machine code and that are efficient in terms of execution speed and storage consumption became a challenging computer science problem. The design and implementation of high-level languages is at the heart of the computer science field called programming languages.

Increasing use of computers in the early 1960s provided the impetus for the development of the first operating systems, which consisted of system-resident software that automatically handled input and output and the execution of programs called “jobs.” The demand for better computational techniques led to a resurgence of interest in numerical methods and their analysis, an activity that expanded so widely that it became known as computational science.

The 1970s and '80s saw the emergence of powerful computer graphics devices, both for scientific modeling and other visual activities. (Computerized graphical devices were introduced in the early 1950s with the display of crude images on paper plots and cathode-ray tube [CRT] screens.) Expensive hardware and the limited availability of software kept the field from growing until the early 1980s, when the computer memory required for bitmap graphics (in which an image is made up of small rectangular pixels) became more affordable. Bitmap technology, together with high-resolution display screens and the development of graphics standards that make software less machine-dependent, has led to the explosive growth of the field. Support for all these activities evolved into the field of computer science known as graphics and visual computing.

Closely related to this field is the design and analysis of systems that interact directly with users who are carrying out various computational tasks. These systems came into wide use during the 1980s and '90s, when line-edited interactions with users were replaced by graphical user interfaces (GUIs). GUI design, which was pioneered by Xerox and was later picked up by Apple (Macintosh) and finally by Microsoft (Windows), is important because it constitutes what people see and do when they interact with a computing device. The design of appropriate user interfaces for all types of users has evolved into the computer science field known as human-computer interaction (HCI).

The field of computer architecture and organization has also evolved dramatically since the first stored-program computers were developed in the 1950s. So called time-sharing systems emerged in the 1960s to allow several users to run programs at the same time from different terminals that were hard-wired to the computer. The 1970s saw the development of the first wide-area computer

networks (WANs) and protocols for transferring information at high speeds between computers separated by large distances. As these activities evolved, they coalesced into the computer science field called networking and communications. A major accomplishment of this field was the development of the Internet.

The idea that instructions, as well as data, could be stored in a computer's memory was critical to fundamental discoveries about the theoretical behaviour of algorithms. That is, questions such as, "What can/cannot be computed?" have been formally addressed using these abstract ideas. These discoveries were the origin of the computer science field known as algorithms and complexity. A key part of this field is the study and application of data structures that are appropriate to different applications. Data structures, along with the development of optimal algorithms for inserting, deleting, and locating data in such structures, are a major concern of computer scientists because they are so heavily used in computer software, most notably in compilers, operating systems, file systems, and search engines.

In the 1960s the invention of magnetic disk storage provided rapid access to data located at an arbitrary place on the disk. This invention led not only to more cleverly designed file systems but also to the development of database and information retrieval systems, which later became essential for storing, retrieving, and transmitting large amounts and wide varieties of data across the Internet. This field of computer science is known as information management.

Another long-term goal of computer science research is the creation of computing machines and robotic devices that can carry out tasks that are typically thought of as requiring human intelligence. Such tasks include moving, seeing, hearing, speaking, understanding natural language, thinking, and even exhibiting human emotions. The computer science field of intelligent systems, originally known as artificial intelligence (AI), actually predates the first electronic computers in the 1940s, although the term artificial intelligence was not coined until 1956.

Three developments in computing in the early part of the 21st century—mobile computing, client-server computing, and computer hacking—contributed to the emergence of three new fields in computer science: platform-based development, parallel and distributed computing, and security and information assurance. Platform-based development is the study of the special needs of mobile devices, their operating systems, and their applications. Parallel and distributed computing concerns the development of architectures and programming languages that support the development of algorithms whose components can run simultaneously and asynchronously (rather than sequentially), in order to make better use of time and space. Security and information assurance deals with the design of computing systems and software that protects the integrity and security of data, as well as the privacy of individuals who are characterized by that data.

Информатика

Информатика – это изучение компьютеров и вычислений, включая их теоретические и алгоритмические основы, аппаратное и программное обеспечение и их использование для обработки информации. Дисциплина информатики включает изучение алгоритмов и структур данных, проектирование компьютеров и сетей, моделирование данных и информационных процессов, а так же искусственный интеллект. Информатика берет некоторые из своих основ из математики и техники, и следовательно, включает в себя методы из таких областей, как теория очередей, теория вероятности и статистика, а также проектирование электронных схем. Информатика также применяет интенсивное использование проверок предположений и экспериментирование вовремя концептуализации, проектирования, измерения и улучшения новых алгоритмов, информационных структур и компьютерных архитектур.

Информатика рассматривается как часть семейства из пяти отдельных, но взаимосвязанных дисциплин: компьютерная техника, информатика, информационные системы, информационные технологии и разработка программного обеспечения. Это семейство стало известным под общим названием как вычислительная техника. Эти пять дисциплин взаимосвязаны в том смысле, что вычислительная техника — это объект их изучения, но они разделены, так как каждая имеет свой собственную перспективу исследований и учебную направленность. (С 1991 года Ассоциация вычислительной техники [ACM], компьютерное сообщество Института инженеров электротехники и электроники [IEEE-CS] и ассоциация Информационных систем [AIS] сотрудничают, чтобы развивать и обновлять систематику этих пяти взаимосвязанных дисциплин и методические рекомендации, которые используют учебные заведения по всему миру для своих программ бакалавриата, магистратуры и исследовательских программ.)

Основные подразделы информатики включают традиционное изучение компьютерной архитектуры, языков программирования и разработки программного обеспечения. Однако, они также включают вычислительную науку (использование алгоритмических техник для моделирования научных данных), графику и визуализацию, взаимодействие человека с компьютером, базы данных и информационные системы, компьютерные сети, а также социальные и профессиональные проблемы, которые уникальны в практике информатики. Как может быть очевидно, некоторые из этих подразделов совпадают с другими современными областями, такими как биоинформатика и вычислительная химия. Эти совпадения являются последствием тенденции среди ученых-информатиков распознавать и действовать на многих междисциплинарных связях в своей области.

Развитие информатики

Информатика возникла как независимая дисциплина в начале 1960-х годов, несмотря на то, что электронно-цифровой компьютер, являющийся

объектом ее изучения, был изобретен примерно двумя десятилетиями ранее. Корни информатики лежат прежде всего в смежных областях математики, электротехники, физики и информационных систем управления.

Математика является источником двух ключевых концепций в развитии компьютеров - идея того, что вся информация может быть представлена как последовательность нулей и единиц и абстрактное понятие «сохраненная программа». В двоичной системе счисления числа представлены как последовательность двоичных цифр 0 и 1 тем же самым способом, как числа в знакомой десятичной системе представлены с использованием цифр от 0 до 9. Относительная простота, с которой два состояния (к примеру, высокое и низкое напряжение) могут быть реализованы в электрических и электронных устройствах, привела естественным образом к двоичной цифре или биту, который стал основной единицей хранения и передачи данных в компьютерной системе.

Электротехника обеспечивает основы проектирования схем, а именно идею о том, что электрические импульсы, поданные на вход схемы, можно комбинировать, используя булеву алгебру, для производства произвольных импульсов на выходе. (Булева алгебра, развитая в 19 веке, предоставила формализм для конструирования схем с двоичными входными значениями из нулей и единиц [ложь или истина, соответственно, в терминологии логики], для получения любой желаемой комбинации нулей и единиц на выходе.) Изобретение транзистора и миниатюризация схем вместе с изобретением электронных, магнитных и оптических носителей для хранения и передачи информации стали результатом достижений электротехники и физики.

Информационные системы управления, изначально названные как системы обработки данных, предоставили ранние идеи, из которых были развиты различные концепции информатики, такие как сортировка, поиск, базы данных, поиск информации и графический интерфейс пользователя. В крупных корпорациях размещались компьютеры, на которых хранилась информация, имеющая центральное значение для ведения бизнеса — платежные ведомости, бухгалтерского учета, управления запасами, производственного контроля, доставки и получения.

Теоретическая работа по вычислимости, которая началась в 1930-х, обеспечили необходимое продвижение этих достижений для проектирования целых машин; ключевым событием стало в 1936 году описание «Машины Тьюринга» (теоретической вычислительной модели, которая выполняет инструкции, представленные как серия нулей и единиц) британским математиком Аланом Тьюрингом и его доказательство вычислительной мощности модели. Еще одним прорывом была концепция компьютера с хранимой программой¹, которую обычно приписывают венгеро-американскому математику Джону фон Нейману. Это изначальные области информатики, которые позже стали известны как архитектура и организация.

В 1950-х большинство компьютерных пользователей работало либо в

1 Архитектура фон Неймана - широко известный принцип совместного хранения команд и данных в памяти компьютера.

исследовательских научных лабораториях, либо в больших корпорациях. Первая группа использовала компьютеры, с целью помочь произвести сложные математические вычисления (к примеру, траектории ракет), пока другая группа использовала компьютеры для управления большими объемами корпоративных данных (к примеру, платежные ведомости и инвентаризационные описи). Обе группы быстро усвоили, что написание программ на машинном языке нулей и единиц не было практичным или надежным. Это открытие привело к развитию языка сборки, который позволил программистам использовать символы для написания инструкций (например, ADD для добавления) и переменных (например, X). Еще одна программа, известная как ассемблер, переводила эти символические программы в эквивалентную двоичную программу, чьи шаги компьютер мог бы выполнить или «исполнить»².

Другими элементами системы программного обеспечения, известные как связывающие загрузчики, были разработаны для комбинирования частей собранного кода и загрузки их в память компьютера, где они могли бы выполняться. Концепция связывания различных частей кода была важна, так как позволяла быть переиспользованными «библиотекам» программ для выполнения общих задач. Это был первый шаг в развитии области информатики называемой разработка программного обеспечения.

Позднее в 1950-х язык сборки был признан таким трудоемким, что в развитии языков высокого уровня (более близких к естественным языкам) началась поддержка более легкого и быстрого программирования. ФОРТРАН появился как главный высокоуровневый язык для научного программирования, в то время как КОБОЛ стал главным языком для программирования в бизнесе. Эти языки принесли с собой необходимость в другом программном обеспечении, называемом компиляторами, которые переводят программы на высокоуровневых языках в машинный код. Так как языки программирования стали более мощными и абстрактными, создание компиляторов, производящих высококачественный машинный код и эффективных с точки зрения скорости исполнения и потребления памяти, стало проблемой информатики, бросающей вызов. Проектирование и реализация высокоуровневых языков находится в центре области информатики, называемой языки программирования.

Увеличение использования компьютеров в начале 1960-х дало толчок к разработке первой операционной системы, которая состояла из системно - резидентного программного обеспечения, автоматически обрабатывающего входные и выходные данные и выполнение программ, называемых «джобсами (jobs)». Потребность в лучших вычислительных техниках привела к возрождению интереса к численным методам и их анализу, к занятию, которое распространилось так широко, что стало известно как вычислительная наука.

В 1970-х и 1980-х годах появились мощные устройства компьютерной

2 В оригинале execute — в семействе Windows большинство файлов программ (исполняемых файлов) имеют расширение exe, сокращение от execute.

графики, как для научного моделирования, так и для других визуальных действий. (Компьютеризированные устройства графики были представлены в начале 1950-х с показом грубых изображений на бумаге и на экранах с электронно-лучевой трубкой [CRT]. Дорогостоящее оборудование и ограниченные возможности программного обеспечения сдерживали эту область от развития до начала 1980-х, когда компьютерная память, необходимая для растровой графики (в которой изображение состоит из маленьких прямоугольных пикселей), стала более доступной. Технология растровой графики, совместно с экранами высокого разрешения и разработкой графических стандартов, делающих программное обеспечение менее зависимым от машин, привели к взрывному росту в этой области. Поддержка всех этих видов деятельности превратилась в область информатики, известную как графика и графические вычисления.

Тесно связанными с этой областью является дизайн и анализ систем, которые взаимодействуют непосредственно с пользователями, выполняющими различные вычислительные задачи. Эти системы пришли к широкому использованию в 1980-х и 1990-х, когда взаимодействие с пользователями через режим командной строки было замещено графическими пользовательскими интерфейсами (ГПИ, ГИП, GUI). Дизайн ГПИ, в котором первопроходцем был Xerox, а позднее подобрал Apple (Macintosh), и, наконец, Microsoft (Windows), важен потому, что он представляет то, что люди видят и делают, когда они взаимодействуют с вычислительным устройством. Проектирование нужных пользовательских интерфейсов для всех типов пользователей превратилось в область информатики, известную как взаимодействие человека и компьютера (HCI).

Область компьютерной архитектуры и организации также значительно изменилась с тех пор, как были разработаны первые компьютеры с хранимыми программами в 1950-х. Так называемые системы разделения времени появились в 1960-х годах, чтобы позволить нескольким пользователям выполнять программы в одно и то же время с разных терминалов, которые были жестко соединены с компьютером проводами. 1970-е увидели разработку первых компьютерных сетей широкого применения (WANs) и протоколов передачи информации на высоких скоростях между компьютерами, разделенными большими расстояниями. По мере развития этих видов деятельности, они слились в область информатики, называемую компьютерные сети и коммуникации. Главным достижением этой области было развитие интернета.

Идея, что инструкции, так же как и данные, могут храниться в памяти компьютера была переломной для фундаментальных открытий о теоретическом поведении алгоритмов. То есть, такие вопросы как «Что может / не может быть вычислено?» были формально рассмотрены с использованием этих абстрактных идей. Эти открытия были первоисточником области информатики, известной как алгоритмы и сложность. Ключевой частью этой области является изучение и применение структур данных, подходящих для различных применений. Структуры

данных, наряду с разработкой наилучших алгоритмов для вставки, удаления и размещения данных в таких структурах, являются главной заботой ученых — компьютерщиков потому, что они (структуры данных) так сильно используются в компьютерном программном обеспечении, особенно в компиляторах, операционных системах, файловых системах и поисковых движках.

В 1960-х годах изобретение накопителей на магнитных дисках обеспечило быстрый доступ к данным, расположенным в произвольном месте на диске. Это изобретение позволило не только к более разумно спроектированным файловым системам, но и к развитию баз данных и систем поиска информации, которые позднее стали неотъемлемой частью для хранения, поиска, и передачи большого количества и разнообразия данных через интернет.

Другой долгосрочной целью исследований информатики является создание вычислительных машин и роботизированных устройств, которые могут выполнять задачи, которые обычно рассматриваются, как требующие интеллекта человека. Такие задачи включают движение, зрение, слух, речь, понимание естественного языка, мышление и даже выражение человеческих эмоций. Область информатики в интеллектуальных системах, изначально известная как искусственный интеллект, в действительности предшествует появлению первых электронных компьютеров в 1940-х годах, хотя термин "искусственный интеллект" не был придуман до 1956 года.

Три разработки в вычислительной технике в начале 21-го века — мобильная техника, клиент-серверные вычисления и взлом компьютеров — способствовали появлению трех новых областей информатики: платформо-зависимая разработка, параллельное и распределенное вычисление, и обеспечение информационной безопасности. Платформо-зависимая разработка - это изучение особых потребностей мобильных устройств, их операционных систем и их приложений. Параллельное и распределенное вычисление касаются разработки архитектур и языков программирования, которые поддерживают разработку алгоритмов, чьи компоненты могут выполняться одновременно и асинхронно (а не последовательно), чтобы лучше использовать времени и пространства. Обеспечение информационной безопасности имеет дело с проектированием вычислительных систем и программного обеспечения, которые обеспечивают целостность и защиту данных, а также приватность частных лиц, которые характеризуются этими данными.