**Unit VI. COMPUTER sOFTWARE. PROGRAMMING LANGUAGES**

*VOCABULARY STUDY*

**Word List**

***Nouns and noun phrases***

control flow **─** [последовательность управляющих команд](https://www.multitran.com/m.exe?s=%D0%BF%D0%BE%D1%81%D0%BB%D0%B5%D0%B4%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8C%D0%BD%D0%BE%D1%81%D1%82%D1%8C+%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D1%8F%D1%8E%D1%89%D0%B8%D1%85+%D0%BA%D0%BE%D0%BC%D0%B0%D0%BD%D0%B4&l1=2&l2=1); computation **─** вычисление; spreadsheet software **─** программное обеспечение для табличных расчётов; freeware **─** бесплатное программное обеспечение; trial period **─**  срок пробного пользования, период тестирования; charge **─** плата за оказываемые услуги; shareware **─** условно-бесплатное программное обеспечение; proprietary software **─** программное обеспечение собственной разработки; paradigm **─** категория; utility **─** служебная программа; script **─** сценарий; essence **─** сущность; back end developer  **─** разработчик серверной части приложения; retail **─** розничная торговля; acronym **─** акроним, вид аббревиатуры, образованной начальными звуками; shortcut ─ быстрая клавиша;

***Adjectives***

non-essential **─** несущественный; instant **─** мгновенный; open-source **─** c открытым исходным кодом; concurrent **─** для использования в многопоточной среде; relevant **─** релевантый, уместный; in-demand **─** востребованный; flexible ─ гибкий;robust **─** устойчивый к сбоям и ошибкам; numerical **─** численный;

***Verbs and adverbs***

Execute **─** выполнять, запускать; ultimately **─** в конечном итоге; cause **─** быть причиной, вызывать; carry out **─** выполнять; edit ─ редактировать; debug ─ отладить, устранить ошибки; purchase ─ покупать; maintain ─ поддерживать в рабочем состоянии; distribute ─ распространять; track ─ отслеживать.

**I. Match the words with the definitions below.**

*Environment, ~~programming language~~, ~~trial period~~, source code, encompass, ~~alter~~, ~~requirement~~, ~~familiar,~~**~~assembler, feature~~.*

1. Testing period of time to discover how effective or suitable something or someone is. trial period

2. Language for writing software. *programming language*

3. The system in which a computer or computer program operates. *Environment*

4. The set of computer instructions that have been written in order to create a program or piece of software. *source code*

5. A typical quality or an important part of something.  *feature*

6. To include several different things.  *encompass*

7. A program that changes computer instructions into machine code (= a set of numbers that gives instructions to a computer). *assembler*

8. An official rule about something that it is necessary to have or to do. *requirement*

9. Easy to recognize because of being seen, met, heard, etc. before. *familiar*

10. To change something, usually slightly, or to cause the characteristics of something to change. *alter*

**II. Using a dictionary add as many words as possible into the table.**

|  |  |  |
| --- | --- | --- |
| ***Verbs***  1. to alter  2. to require  3. to execute  4. to originate  5. to interpret  6. to assemble  7. to access | ***Adjectives***  Altered/ alterable  required  executable  original  interpretable  assembled  accessible | ***Nouns***  alteration  requirement  execution  origin  interpreter  assembly  accessibility |

**III. Choose the words with similar meaning from the two groups and arrange them in pairs.**

A. Load, debug, end-user, open-source, pass, provide, translate, programming language.

B. Application, software, programming language, editor, instruction, environment, software.

End-user software, open-source editor, provide environment, translate programming language, load application, debug software, programming language instruction.

**IV. Complete the sentences with the word collocations below that describe the advantages of high-level languages.**

*Easy to detect and remove errors; Machine Independence; Built-in library functions; Easy to understand; Easy to learn; Easy to write program.*

There are several advantages of high-level programming languages. The most important advantages are:

a) *Easy to learn* - the high-level languages are easier to learn than low level languages. The statements written for the program are similar to English-like statements.

b) *Easy to understand* - the program written in high level language by one programmer can easily be understood by another because the program instructions are similar to the English language.

c) *Easy to write program* - in high level language, a new program can easily be written in a very short time. The larger and complicated software can be developed in few days or months.

d) *Easy to detect and remove errors* - the errors in a program can be easily detected and removed. mostly the errors are occurred during the compilation of a new program.

e) *Built-in library functions* - Each high-level language provides a large number of built-in functions or procedures that can be used to perform specific task during designing of new programs. In this way, a large amount of time of programmer is saved.

f) *Machine Independence* - program written in a high-level language is machine independent. It means that a program written in one type of computer can be executed on another type of computer.

**V. Make up your own sentences using the following words and word combinations.**

*A collection of computer programs, to execute the software, high-level programming languages, different classes of computer software, provide an environment or platform, test, debug, compiler, interpreter, assembler, Freeware, Shareware, Open Source Software, Closed Source Software, Utility Software.*

**полина**

Computer software is a general term used to describe *a collection of computer programs*.

The computer is able to *execute the software*.

*High-level programming languages* is a Java, C, C++, PHP, Delphi

*Different classes of computer software* are divided into *System Software*, *Application Software* , *Programming software*.

*Provide an environment or platform* for software development.

It is better to *test* the software before sending it to use.

*Debugging* is an important part of creating software.

The compiler translates the text written in the programming language into a set of machine codes.

Translators can be compilers, *interpreters* and assemblers.

*Assembler* is a low-level machine-oriented programming language.

*Freeware software* is any software that is available to use for free.

The *shareware* program provides a trial period of free use.

*Open Source Software* is a type of software that has an open-source code that is available to use for all users.

*Closed Source Software*  are the types of software that are non-free for the programmers.

*Utility software* is considered a subgroup of system software.

**Кристина**

1. The software consists of *A collection of computer programs.*
2. Once **the** **software** is loaded, the computer is able to **execute** **the** **software**.
3. **high**-**level** **programming** **languages** are a **programming** **languages** with strong abstraction from the details of the computer.

3. now we are learning such a high-level language c++

4. There are different classes of software, for example, system (also called general) and application (called special).

5. sites that provide a platform for creating websites have different adaptive templates

6. to identify errors in a software product, it is necessary to test it

7. we use debug to execute the program in step-by-step mode

7. during debugging, memory and register windows, disassembled code, call stack, control and local values are available

8. A compiler is a program that translates text written in a programming language into a set of machine codes

8. there are different types of compilers, for example, Interpretive, Incremental, dialog, debugging, and so on

9. An interpreter is a kind of translator that performs interpretation

9. The interpreter line-by-line analyzes, processes and executes the source code of the program or query

10. In computer programming, **assembler** is any low-level programming language in which there is a very strong correspondence between the instructions in the language and the architecture's machine code instructions.

10. visual studio provides an opportunity to view the program code in assembly language

11. Antiviruses and programs distributed as Freeware are designed for home users and are free for private use.

12. **Shareware** is a type of proprietary software which is initially provided free of charge to users, who are allowed and encouraged to make and **share** copies of the program.

13. The source code of the Open Source Software is available for review, study and modification

14. closed-source software, is computer software for which the software's publisher reserves some licensing rights to use, modify, share modifications, or share the software,

14. Closed Source Software is the opposite of open-source or free software.

15. Utility software is software designed to help to analyze, configure, optimize or maintain a computer.

15. Utility software is used to support the computer infrastructure

**VI. Translate into English.**

Программное обеспечение – наряду с аппаратными средствами, важнейшая составляющая информационных технологий, включающая компьютерные программы и данные, предназначенные для решения определенных задач.

В компьютерном жаргоне часто используется слово «софт» от английского software, которое в этом смысле впервые применил в журнале American mathematical Monthly математик из Принстонского университета Джон Тьюки (John W. Tukey) в 1958 г.

По назначению ПО разделяется на системное, прикладное и инструментальное. Системное программное обеспечение реализует связь аппаратного и программного обеспечения, выступая в качестве «межслойного интерфейса», с одной стороны которого аппаратура, а с другой приложения пользователя. Прикладное программное обеспечение призвано решать прикладные задачи пользователя. Например, финансовое управление, сеть поставок, управление транспортными расходами. Инструментальное программное обеспечение предназначено для использования в ходе проектирования, разработки и сопровождения программ. Виды инструментального ПО: текстовые редакторы, компиляторы, интерпретаторы, линковщики, отладчики, ассемблеры.

По способу распространения (доставки, оплаты, ограничения в использовании) ПО подразделяются на:

Freeware – бесплатное ПО, являющееся собственническим. Условия его распространения могут запрещать его копировать, изменять, распространять;

Shareware – условно-бесплатное ПО. Пользователю предлагают пробную версию с напоминанием о необходимости оплаты использования программы. Основной принцип этого ПО: «попробуй, прежде чем купить»;

Proprietary ware – собственническое ПО. Правообладатель сохраняет за собой монополию на его использование, копирование, модификацию.

Хуй:

Software is, along with hardware, the most important component of information technology, including computer programs and data designed to solve certain tasks.

In computer jargon, the word "software" is often used from the English software, which in this sense was first used in the journal American mathematical Monthly by a mathematician from Princeton University John W. Tukey in 1958.

According to its purpose, the software is divided into system, application and instrumental. The system software implements the communication of hardware and software, acting as an "interlayer interface", on one side of which is the hardware, and on the other the user's applications. The application software is designed to solve the user's application tasks. For example, financial management, supply chain, transportation cost management. The tool software is intended for use during the design, development and maintenance of programs. Types of tool software: text editors, compilers, interpreters, linkers, debuggers, assemblers.

By distribution method (delivery, payment, usage restrictions) Software is divided into:

Freeware – free proprietary software. The terms of its distribution may prohibit its copying, modification, distribution;

Shareware is shareware. The user is offered a trial version with a reminder of the need to pay for the use of the program. The basic principle of this software is: "try before you buy";

Proprietary ware is proprietary software. The copyright holder reserves the monopoly on its use, copying, modification.

*A. TEXT STUDY*

**I. Read the text and answer the following questions.**

1. What does software include?

2. What does software execution mean?

3. What operations do computers carry out?

4. Who introduced the term ‘software’?

5. What does system software coordinate?

6. Give examples of application software.

7. What software is used for creating both the system as well as application software?

8. Characterize some additional subcategories of software.

**Text A.** **Different Classes and Types of Computer Software**

*Computer software* is a general term used to describe a collection of computer programs, procedures and documentation that perform some tasks on a computer system. Software includes websites, programs, video games, etc. that are coded by programming languages like C, C++, etc. The term “software” is sometimes used in a broader context to mean anything which is not hardware (which encompasses the physical interconnections and devices required to store and execute (or run) the software) but which is used with hardware, such as film, tapes, records, etc.

In computers, software is loaded into RAM and executed in the CPU. Once the software is loaded, the computer is able to execute the software. This involves passing instructions from the application software, through the system software, to the hardware which ultimately receives the instruction as machine code. Each instruction causes the computer to carry out an operation – moving data, carrying out a computation, or altering the control flow of instructions. Software is usually written in high-level programming languages that are easier and more efficient for humans to use (closer to natural language) than machine language. The term “software” was first used by John. W. Tukey in 1958.

A software is a collection of programs that help one to communicate with the hardware of the computer. There are different classes of computer software which are useful for several purposes.

*System Software* coordinates the complete system hardware and provides an environment or platform for all the other types of software to work in. It is the most basic type of software in any computer system, which is essential for other programs, applications and indeed for the whole computer system to function. For desktop computers, laptops and tablets System software examples are Microsoft Windows 10, Mac OS, Linux, Ubuntu, devices drivers, etc. For smartphones: Apple’s iOS, Google’s Android, Windows Phone OS.

*Application Software* comprises those programs that help the user perform tasks of his/ her choice. They are non-essential software which are installed and run depending upon the requirements, in the environment provided by the system software. Application software examples are MS Office, Open Office, Media Players, educational software, media development software, antivirus software, etc.

There are some examples of application software that allow you to do specific work: **MS Excel:** It is spreadsheet software that you can use for presenting and analyzing data. **Photoshop:** It is a photo editing application software by Adobe. You can use it to visually enhance, catalogue and share your pictures. **Skype:** It is an online communication app that you can use for video chat, voice calling and instant messaging.

*Programming software* is used to write, test, debug, and develop other software programs and applications. They are used for creating both the system as well as application software.

Programming software is used by software programmers as translator programs. They are facilitator software used to translate programming languages (i.e., Java, C++, Python, PHP, BASIC) into machine language code. Translators can be compilers, interpreters and assemblers. You can understand compilers as programs that translate the whole source code into machine code and execute it. Interpreters run the source code as the program is run line by line. And assemblers translate the basic computer instructions – assembly code – into machine code.

Different programming language editors, debuggers, compilers and Integrated Development Environments (IDE) are examples of programming software. For example: Eclipse – a Java language editor; Coda – programming language editor for Mac; Notepad++ – an open-source editor for Windows; Sublime Text – a cross-platform code editor for Mac, Windows, and Linux.

There are **five additional subcategories of software**. These are: Freeware; Shareware; Open Source Software; Closed Source Software; Utility Software.

*Freeware software* is any software that is available to use for free. They can be downloaded and installed over the internet without any cost. Some well-known examples of freeware are: Google Chrome; Skype; Instagram; Snapchat; Adobe reader. Although they all fall under the category of Application or end-user software, they can further be categorized as freeware because they are free for you to use.

*Shareware*, on the other hand, are software applications that are paid programs, but are made available for free for a limited period of time known as ‘trial period’. You can use the software without any charges for the trial period but you will be asked to purchase it for use after the trial ends. Shareware allows you to test drive the software before you actually invest in purchasing it. Some examples of Shareware that you must be familiar with are: Adobe PhotoShop; Adobe Illustrator; Netflix App; Matlab; McAfee Antivirus.

*Open Source Software* is a type of software that has an open-source code that is available to use for all users. It can be modified and shared with anyone for any purpose. Common examples of open source software used by programmers are: LibreOffice; PHP; GNU Image Manipulation Program (GIMP).

*Closed Source Software.* These are the types of software that are non-free for the programmers. For this software, the source code is the intellectual property of software publishers. It is also called ‘proprietary software’ since only the original authors can copy, modify and share the software. Following are some of the most common examples of closed-source software: .NET; Java; Android; Microsoft Office; Adobe PhotoShop.

*Utility software* is considered a subgroup of system software. They manage the performance of your hardware and application software installed on your computer, to ensure they work optimally. Some features of utility software include: Antivirus and security software; File compressor; Disk cleaner; Disk defragmentation software; Data backup software.

There can be multiple ways to classify different types of computer software. The software can be categorized based on the function they perform such as Application software, System software, Programming Software, and Driver software. They can also be classified based on different features such as the nature of source code, accessibility, and cost of usage.

**II. Study the table, analyze and describe different types of software.**

| **Application Software Type** | **Examples** |
| --- | --- |
| **Word processing software:** Tools that are used to create word sheets and type documents etc. | Microsoft Word, WordPad, AppleWorks and Notepad |
| **Spreadsheet software:** Software used to compute quantitative data. | Apple Numbers, Microsoft Excel and Quattro Pro |
| **Database software:** Used to store data and sort information. | Oracle, MS Access and FileMaker Pro |
| **Application Suites:** A collection of related programs sold as a package. | OpenOffice, Microsoft Office |
| **Multimedia software:** Tools used for a mixture of audio, video, image and text content. | Real Player, Media Player |
| **Communication Software:** Tools that connect systems and allow text, audio, and video-based communication. | MS NetMeeting, IRC, ICQ |
| **Internet Browsers:** Used to access and view websites. | Netscape Navigator, Microsoft Edge, and Google Chrome |
| **Email Programs:** Software used for emailing. | Microsoft Outlook, Gmail, Apple Mail |

**III. Choose the best option to the following statements.**

1. In computers, software is loaded into … and executed in the CPU.

a) ROM

b) RAM

c) CPU

2. Each instruction causes the computer to … an operation.

a) execute

b) promote

c) change

3. Software is usually written in … languages.

a) mark up

b) assembler

c) high-level

4. The software can be categorized based on … .

a) different features such as the nature of source code, accessibility, etc.

b) the function they perform.

c) all of the above.

5. … can be modified and shared with anyone for any purpose.

a) Utility software

b) Shareware

c) Open source software

**IV. Choose the correct word to fill the spaces.**

1. Turn on your computer. It will usually take a few minutes to \_\_\_\_\_\_\_\_\_\_.

a. boot itself b. boot up c. get booted

2. Windows XP, Macintosh OSX and Linux are \_\_\_\_\_\_\_\_\_\_.

a. operating systems b. operating tools c. operators

3. On my computer, I have a picture of my cat as the \_\_\_\_\_\_\_\_\_\_.

a. desktop background b. desktop picture c. desktop scene

4. Microsoft Word, Adobe Acrobat and CorelDraw are programs or \_\_\_\_\_\_\_\_\_\_.

a. applicators b. appliers c. applications

5. To open Microsoft Word, click on the \_\_\_\_\_\_\_\_\_\_.

a. picture b. symbol c. icon

6. I keep all my digital photos in a \_\_\_\_\_\_\_\_\_\_ called "Photos".

a. folder b. packet c. box

7. Is it possible to open Microsoft Excel \_\_\_\_\_\_\_\_\_\_ in Word?

a. texts b. files c. pages

8. In Microsoft Word, to start typing a new letter, open a new \_\_\_\_\_\_\_\_\_\_\_.

a. document b. page c. paper

9. When you \_\_\_\_\_\_\_\_\_\_ a document, it's sent to the recycle bin.

a. destroy b. erase c. delete

10. Deleted documents stay in the recycle bin until you \_\_\_\_\_\_\_\_\_\_ it.

a. wash b. empty c. clean

11. In Windows, the icon is just a \_\_\_\_\_\_\_\_\_\_ to the application. If you delete the icon, the application will still be on your computer.

a. connector b. shortcut c. link

12. If the computer crashes, you can try pressing the \_\_\_\_\_\_\_\_\_\_ button.

a. restart b. recommence c. replay

13. When I've finished using my computer, I always \_\_\_\_\_\_\_\_\_\_.

a. close it down b. shut it down c. shut it off

14. If I leave my computer on without using it, after a while it goes into \_\_\_\_\_\_\_\_\_\_ mode.

a. stand down b. waiting c. standby

*B. TEXT STUDY*

**I. Read the text and answer the following questions.**

1. What is a programming language used for?

2. What are the types of programming languages?

3. Characterize some of the most relevant and in-demand languages?

**Types of programming languages**

A programming language is a computer language that a programmer uses to develop software programs, scripts, or other sets of instructions for computers to execute.

Although many languages share similarities, each has its own syntax. Once a programmer learns the languages rules, syntax, and structure, they write the source code in a text editor or IDE. Then, the programmer often compiles the code into machine language that can be understood by the computer. Scripting languages, which do not require a compiler, use an interpreter to execute the script.

Each of the different programming languages mentioned can be broken into one or more of the following types (or paradigms) of languages.

High-level (most common) / low-level; Declarative / imperative / procedural; General-purpose / domain-specific; Object-oriented / concurrent; Command / Compiled / Script language; Answer set. A language can also be broken into one of five generation languages. Today, there are hundreds of different programming languages.

*Application and program development* involve programs you work with on a daily basis. For example, the Internet browser you are using to view this web page is considered a program. If you are interested in developing a program, consider the following languages: C, C#, C++, Java, Swift, Visual Basic.

*Artificial intelligence* or related fields involve creating the character interactions in computer games, portions of programs that make decisions, chatbots, and more. If you're interested in developing an AI, consider the following languages: AIML, C, C#, C++, Prolog, Python.

*Database* developers create and maintain databases. If you're interested in creating or maintaining a database, consider any of these: DBASE, FoxPro, MySQL, SQL, Visual FoxPro.

*Game development* involves creating computer games or other entertainment software. If you're interested in developing a game, consider these: C, C#, C++, DarkBASIC, Java.

*Computer drivers* and programming hardware interface support are a necessity for hardware functionality. If you're interested in developing drivers or software interfaces for hardware devices, consider these: Assembly, C.

*Internet and web page development* are the essence of the Internet. Without developers, the Internet would not exist. If you're interested in creating web pages, Internet applications, or other Internet-related tasks, consider these languages: HDML, HTML, Java, JavaScript, Perl, PHP, Python, XML.

There are dozens of programming languages used in the industry today. Let us overview some most important, relevant and in-demand of these languages.

**Python** is an advanced programming language that is interpreted, object-oriented and built on flexible and robust semantics. Python was developed in the late 1980s in the Netherlands and first released to the public in 1991. Python is used by developers, software engineers, back end developers, Python programmers, also by employers in information technology, engineering, professional services and design in scientific and numeric computing; desktop graphical user interfaces (GUIs). Python lets you work quickly to integrate systems as a scripting or glue language. It’s also suited for Rapid Application Develop (RAD). NASA uses Python in its Integrated Planning System as a standard scripting language. It is simple to learn and easily read.

**Java** is a general-purpose, object-oriented, high-level programming language with several features that make it ideal for web-based development. 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. It is used by software engineers, Java developers, by employers in communications, education, finance, health sciences, hospitality, retail, Internet of Things (IoT), Cloud Computing, etc.

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. Its features are application portability, robust and interpreted language, extensive network library.

**Ruby** is an open-sourced, object-oriented scripting language that can be used independently or as part of the Ruby on Rails web framework. Designed in 1995, it is used by software engineers, data science engineers, employers in technology, engineering, professional services, design, science and quality control, Web App Development, Robotics, Networking. Ruby is used for simulations, 3D modeling, and to manage and track information. Amazon, Twitter were created using Ruby on Rails. It is free to use, copy, modify and distribute.

**HTML** is the standard markup language used to create web pages; it ensures proper formatting of text and images (using tags) so that Internet browsers can display them in the ways they were intended to look. HTML was created by physicist Tim Berners-Lee in 1990 to allow scientists to share documents online. Is used by Web developers, technical editors, email designers, software engineers. HTML is used to create electronic documents (pages) displayed online. Visit any page and you will see an example of HTML in action. The diversity and complexity in the structure and appearance of today’s sites is made possible with HTML. It is free and accessible, multiple versions are available.

**JavaScript** is a client-side programming language that runs inside a client browser and processes commands on a computer rather than a server. Despite its name, JavaScript is not related to Java. JavaScript was designed by Netscape and originally known as LiveScript, before becoming JavaScript in 1995. It used by Khan Academy, Linkedin, Yahoo, etc. 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.

**C Language** is a structure-oriented, middle-level programming language mostly used to develop low-level applications. It was developed in 1972 at Bell Labs specifically for implementing the UNIX system. It eventually gave rise to many advanced programming languages, including C++, Java C#, JavaScript and Pearl. It is used by employers in Microsoft, Apple, Oracle, in Embedded Systems, Systems Programming, Artificial Intelligence, Industrial Automation, Computer Graphics, Space Research, Image Processing and Game Programming. C Language is used to develop systems applications that are integrated into operating systems such as Windows, UNIX and Linux, as well as embedded software. Applications include graphics packages, word processors, spreadsheets, operating system development, database systems, compilers and assemblers, network drivers and interpreters.

**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.

**C#** (pronounced C-sharp) is a multi-paradigm programming language that features strong typing, imperative, declarative, functional, generic, object-oriented and component-oriented disciplines. C# helps developers create XML web services and Microsoft .NET-connected applications for Windows operating systems and the internet. It is similar to Java in capabilities and ideal for beginners.

**PHP** (Hypertext Preprocessor) is an open-source scripting language designed for creating dynamic web pages that effectively work with databases. It is also used as a general-purpose programming language.

**SQL** (Structured Query Language) is a database query language that allows adding, accessing and managing content in a database. It is the language that allows programmers to perform the common acronym CRUD (Create; Read; Update; Delete) within a database.

**Swift** is Apple’s newest open-source, multi-paradigm programming language for iOS and OS X apps. Swift integrates Objective-C’s named parameters and object-oriented model, while including an advanced compiler, debugger and framework infrastructure.

**II. Comprehension Check. State whether the statements are true or false. Correct if necessary.**

1. When a programmer learns the languages rules, syntax, and structure, he/she can write a programming language in a text editor or IDE.

2. Scripting languages use a compiler as well as an interpreter to execute the script.

3. Database developers involve creating the character interactions in computer games.

4. Python lets you work quickly to integrate systems as a scripting or glue language.

5. Java was developed according to the principle of interchangeability.

6. C++ was developed in 1972 at Bell Labs specifically for implementing the UNIX system.

7. C# helps developers create XML web services and Microsoft .NET-connected applications for Mac OS.

8. Swift is Apple’s newest open-source, multi-paradigm programming language for iOS and OS X apps.

**III. Insert the missing words.**

*Close, drag and drop, find, free up, installed, launch, password, renamed, running, save, search, start menu, uninstalling, user, window.*

1. I couldn't open the document you emailed me. I don't have Microsoft Word \_\_\_\_\_\_\_\_\_\_\_\_ on my computer.

2. Click on that icon to \_\_\_\_\_\_\_\_ Google Chrome.

3. I \_\_\_\_\_\_\_\_\_ an important document, and now I can't find it.

4. If your computer is \_\_\_\_\_\_ several applications at the same time, it's more likely to crash. It's better to \_\_\_\_\_\_\_\_\_ the applications. you're not using.

5. You can access all the applications on your computer from the \_\_\_\_\_\_\_\_\_.

6. You can view two Word documents on the screen at the same time. You just open a new \_\_\_\_\_\_\_\_.

7. It's easy to move files into a folder. You can just \_\_\_\_\_\_\_\_\_\_\_\_.

8. I asked the computer to \_\_\_\_\_\_\_\_for files with "English" in the name, but it didn't \_\_\_\_\_\_\_\_\_\_\_\_ any.

9. This is a shared computer. Each \_\_\_\_\_\_\_\_\_ has their own \_\_\_\_\_\_\_\_.

10. You can \_\_\_ space on your hard drive by \_\_\_\_ applications you never use.

11. If you \_\_\_ your photos as JPEGs instead of TIFFs, you'll use a lot less memory.

**IV. Match the words with the punctuation marks and symbols.**

1. Full stop 2. comma 3. exclamation mark 4. question mark 5. single quotes 6. double quotes 7. dollar sign 8. percentage sign 9. ampersand 10. asterisk 11. hash 12. brackets 13. left bracket 14. square brackets 15. underscore 16. hyphen 17. plus sign 18. equals sign 19. colon 20. semicolon 21. "at" sig 22. forward slash 23. backward slash 24. Arrow

a. ! b. @ c. , d. & e. . f. = g. ‘Hello’ h. → i. \*j. “Hello” k. \_ l. - m. ? n. / o. ( ) p. $ q. \ r. [ ] s. % t. ( u. # v. : w. + x. ;

**V**. **Speak about different types of computer software and programming languages.**