PROGRAMMING IN ASSEMBLER

Berdiyev Alisher (PhD)

Teacher of the Department, "Systematic and Practical Programming", Tashkent University of Information Technologies named after Muhammad Al-Khwarizmi, UZBEKISTAN

Xalikova Madina Shukhratovna

(Tashkent, Uzbekistan) Tashkent University of Information Technologies named after Muhammad Al-Khorazmi

Faculty: Radio and mobile communication, 3-rd year of bachelor

xoliqovamadina78@gmail.com

phone:+998998989262

Annotation. The user is also required to know the computer "language" to communicate with the computer. In computer understanding, a "language" is called a programming language. In order to solve an issue on a computer, first of all, it will be necessary that its algorithm be written according to the structure and the indications and rules that the computer understands this algorithm. It turns out that the recording consists of a consistent arrangement of instructions that the program can perform. The process of building a program for a computer is programming, and the person who builds the program is called a programmer. The basic principles of character-coded languages are that machine codes are marked with characters corresponding to them, and automatic memory allocation and error handling are included. Such a machine received the name of an adapted language - assembly language. Typically programming is done on a high-level programming languages (Delphi, Java, C++, Python) tool. Due to the proximity of the semantics of these programming languages to the human language, the process of building a program is much easier. This article shows the assembly programming language, its content essence and its place in the present day.

Keywords: Assembler, programming, computer, operatsio system, RAM

INTRODUCTION

Programming on a computer is-giving different commands for a computer microprocessor, giving commands about where to change and what to enter or output-in the computer world there are many programming languages, and the number of people interested in programming and it is growing. Programs that do the same type of work can be written in Basic, Pascal, Ci and other languages. Pascal, Fortran, and Kobol are universal languages. The closer an algorithmic language is

to human languages, the higher the language is called. Machine language is the lowest-level language. Machine language consists of these numbers, for example: 010110100010101

Programming languages are divided into 2 large groups. Lower and upper level programming language. The lower-level programming language is much more complex-they are used in very special areas, and their specialists are also very few. Because lower programming languages (e.g.: assembler) may often be needed when working with microprocessors. A high-level programming language is commonly used for various programming jobs. EHM (electronic computing machine) is now included in programming at the time of occurrence, only in machine languages, that is, in the codes of actions that EHM must perform using numbers. For the machine in this case, there were 2-lik, 6-lik, 8-lik number systems as the system. The program is entered by means of numbers in this number system. In high-level programming, languages in machine-adapted (directed) character codes are considered as opposed to machine languages. The assembly language has a special place in programming. The assembly language program is compiled directly using the central processor commands. This helps the programmer to truly know the capabilities of the processor. As noted above, from the language of the Assembly mainly computer devices. the program that works directly with and manages them uses in the construction of theiar (drivers). For example, a control program that is written for all, from automated robots in machine-making enterprises to electronic medical devices, is one of these. But for all currently available devices, the controller application has been written, now the question may arise as to what it is necessary to study the assembler. True, but life does not stop at the moment with the available computer capabilities. Day by day, new devices are being developed and Assembly programmers will be needed to build programs that manage them.

MAIN PART

There is also another dominant aspect of getting a programmer to know Assembler even when it does not need programming. The fact is that the code of existing programs, the code of which is hidden, can only be restored in the assembly language. Knowing the assembler, on the other hand, helps to learn the recovered code and make changes. Another achievement of Assembly is that the program written in it works much faster than in other tlls. Therefore, the assembler can be used not only in the construction of programs that control computer devices, but also in the construction of a program that solves the general issue.

Hence, in the following cases in the plural, the program is written in assembler:

• Computer devices control software when needed porridge.
 • When writing the program part, which should work quickly.
• When writing parts of the operating system that work with devices.
• Computer virussinl when building, studying ulami and writing antiviruslami against them.

The word Assembler is derived from English, meaning collector/ collector. Programs written in assembler are written through text editors as plain text, as in all programming languages. In fact, an assembler is also a program that converts Assembly commands in a given text to machine commands. The reason this language is called a lower-level language is because assembly language looks very close to machine language. Each task, task, action in it (by deed, the commands that perform the arithmetic action are envisaged) or command corresponds to a specific machine-level command, that is, each command in the Assembly has its own unique machine code, respectively. For example, in a specific location in high-level programming languages, or the command is represented by 10-15 close machine commands in a machine language. That is why assembler is distinguished from other programming language compilers by its simplicity.

Assembler (Eng. "Assembler") is a low-level programming language that is an intermediate link between machine code and high-level programming languages. It interacts directly with the hardware and is used to write programs that control a computer or other devices at a lower level. Code written in this language is usually stored using the ASM extension.

Assembly programs are written as a set of mnemonic instructions, each corresponding to a specific processor command. These instructions are then translated (assembled) into machine code — a set of binary numbers that the central processor understands and performs the corresponding operations. Assuming machine language is the first generation, Assembler is the second generation language. It interacts directly with the processor, and each of its commands is instructions for the processor, and not for the operating system or file system. The process of translating assembly language into machine code is called assembly.

Assembly commands consist of operation codes and operands. Operands are addresses where the processor receives data for computation and places the result. These addresses can be RAM cells or registers-the internal memory of the processor. The processor runs faster with registers than Ram.

Assembler language operations have a mnemonic form that is easy to remember:

Add-add (from English. addition);

"Innovations in Science and Technologies" scientific electronic journal
www.innoist.uz
ISSN : 3030–3451
Volume 1
№ 1
February, 2024

SUB-subtraction(Eng. subtraction;

Mule-multiplication (Eng. multiplication), etc.

Registers and memory cells take symbolic names, such as:

Names for EAX, EBX, AX, ah — registers;

mem1 is the name of a memory cell.

An example of a command to add numbers from registers Ax and bx: add ax, BX and here the command to subtract numbers from registers ax and BX: sub ax, bx

There are also directives in assembly language-there are compiler control commands-some of them are:

INCLUDE-open the file and start compiling it;

EXIT-stop compiling the file;

To give the DEF-register a symbolic name, etc.

RESEARCH METHODOLOGY

Do not think that an Assembler is just a set of processor instructions with a convenient entry for the programmer. It is a complete programming language where cycles, conditional operators, procedures and functions can be implemented. For example, we write the word "Hello, World":
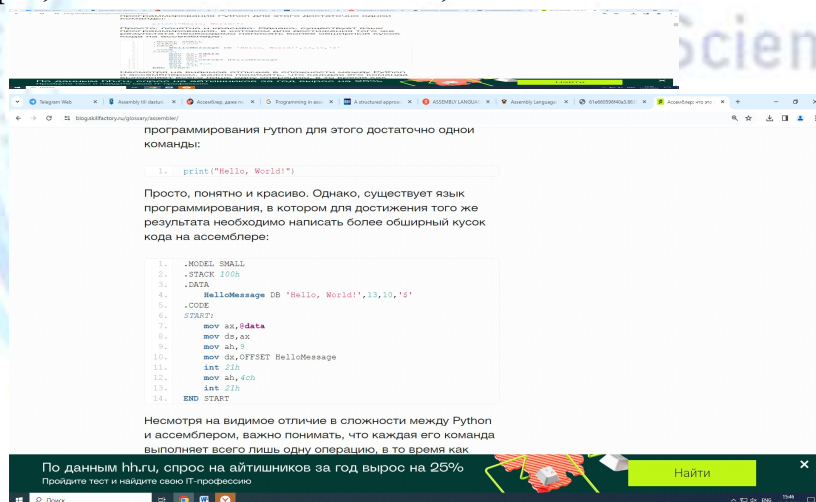
Here is an example of assembly code that shows numbers from 1 to 10:
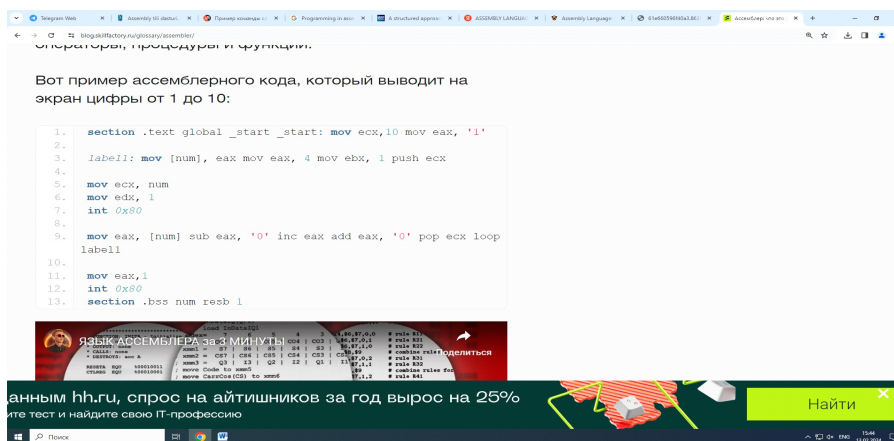
Assembler, even by our example "Hello, world" shows that it is not as development-friendly as high-level languages. No one writes big programs in this language, but there are places where it becomes indispensable:

Development of built-in applications for microcontrollers. Microcontrollers are installed on alarm systems, control panels, sensors, household appliances, modems and other devices. Microcontrollers are even used in robotics and satellite navigation systems. Their memory is limited, so the assembler is ideal for programming them, since each of its commands is transmitted to one command in binary code. This allows you to determine the running time of the program and the memory required to save it, based on the original text of the program.

Write device drivers and some components of operating systems, for example, a kernel or bootloader. Some amateur operating systems, such as Menuetos and KolibriOS, are written in full assembly code. In addition, it can be found in applications for game consoles and multimedia codecs.

Reverse engineering-reverse development of programs. This field is used to understand the operation of programs and their algorithms in cases where there is no source code for any reason. This is done by antivirus companies that study viruses and Trojans, creators of drivers and operating systems, and just curious people. In addition, computer attackers actively use reverse engineering for program hacking, vulnerability searches, virus typing, key generation, and other illegal actions.

So, if you are interested in developing new microprocessors or engaged in reverse engineering, then it is worth seriously studying the assembly language.

CURRENTLY, Assembly programmers are in demand, although on job search websites you cannot find a job with the need for the same language programmer. But there are many announcements that require language proficiency in addition to high-level languages such as C, C++ or Python. Vacancies can be related to reverse engineering, computer security, development of drivers and

applications for microcontrollers/microprocessors, system applications and other areas.

REFERENSES

1.   Abasxanova, X. Yu. "Textbook for college students majoring in" digital technology" 5.55. 01.01-" Telecommunication technologies" Tashkent." (2021).

2.   Abaskhanova, X. Yu, and l. N. Juraev. "Khoshimova FR Textbook for college students majoring in" digital technology" 5.55. 01.01-" Telecommunication technologies". Tashkent." (2021).

3.   Abaskhanova, H. Yu, and K. Sherjanova. "Creating microprocessor systems for people with disabilities creating microprocessor systems for people with disabilities. Collection of reports of the republican scientific-practical conference" prospects for the development of information and communication technologies." Karshi-2018." 87-89.

4.   Abaskhanova, H. Y. "Applying infocommunication technologies to agriculture." Current problems of modern science. Xalqaro конференция. Chicago USA-2022.-б.

5.   Yu, Abaskhanova Kh. "Analysis of information and communication technologies in green environment monitoring." International conference on information science and communications technologies applications, trends and opportunies: ICISCT. 2022.

6.   Yu, Abaskhanova KH. "Features of introduction of innovative technologies in agriculture of Uzbekistan." International scientific journal "Universum: Technical science (2021).

7.   Tursunaliyevich, F. N. ., & Sharofiddinovich, S. S. . (2023). Barqaror Rivojlanishni Maktabgacha Ta'lim Yoshidagi Bolalarga Singdirish. *JOURNAL OF SCIENCE, RESEARCH AND TEACHING*, *2*(2), 26–29. Retrieved from https://jsrt.innovascience.uz/index.php/jsrt/article/view/45

8.   Yu, Abaskhanova Kh. "Advantages of using digital technologies in agriculture." Agro science. Agrarian-economic, scientific-practical magazine. Tashkent-2022.3-appendix (81)-no.-pages104-105.

9.   Abaskhanova K. Y. ORGANIZING THE CLASSES OF MICROCONTROLLER SYSTEMS AT SCHOOLS. – 2023.

10.   Abaskhanova K. Y. USE OF INNOVATIVE TECHNOLOGIES IN TEACHING THE SCIENCE OF PROGRAMMABLE DIGITAL DEVICES. – 2023.

11.   Abasxanova X , Ruziyeva Soxibjamol The importance of innovative technologies in remote monitoring of agricultural products. 2024