

Увод в програмирането

Въведение

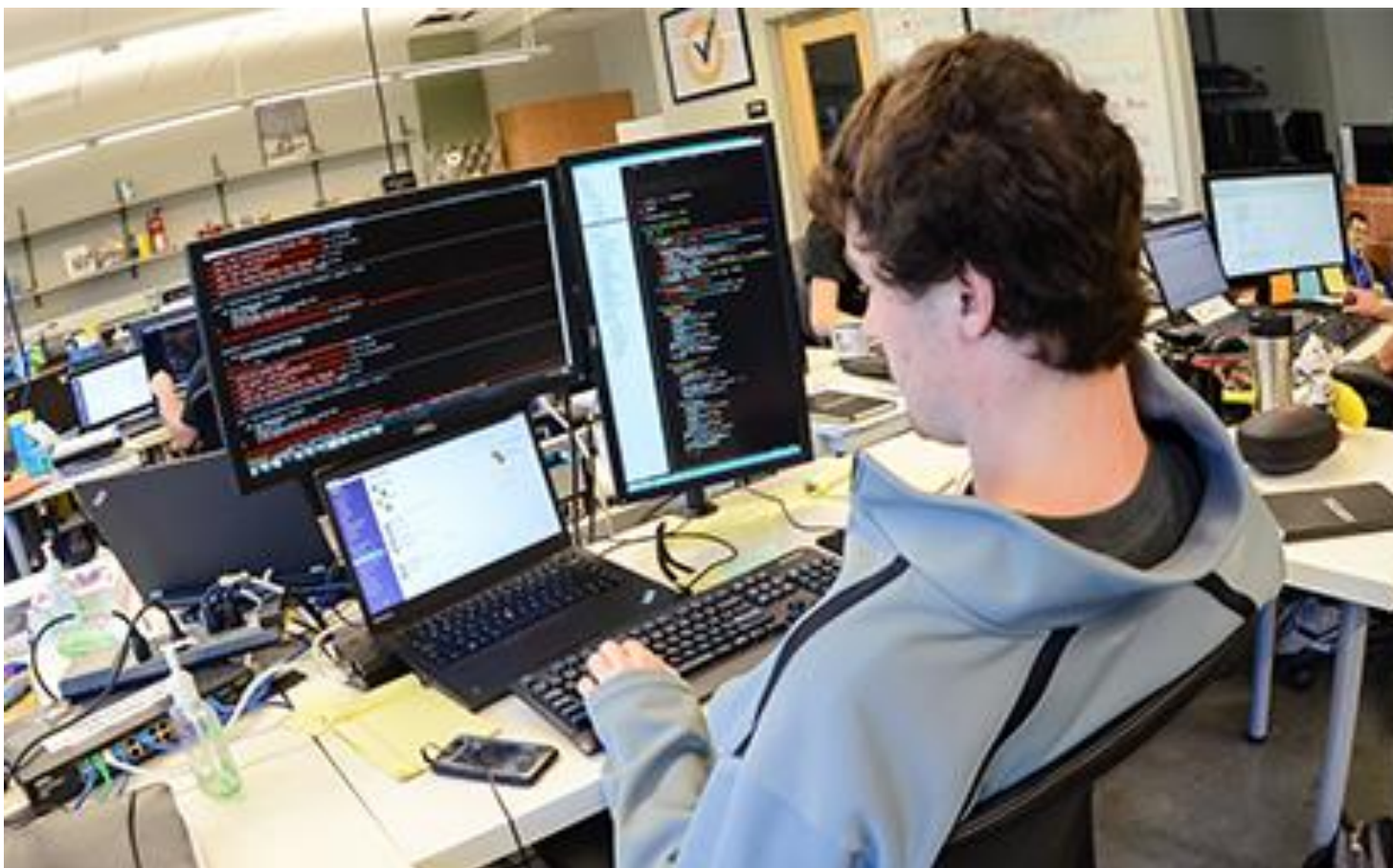
2017-2018 г.

ФМИ, специалност „Софтуерно инженерство“

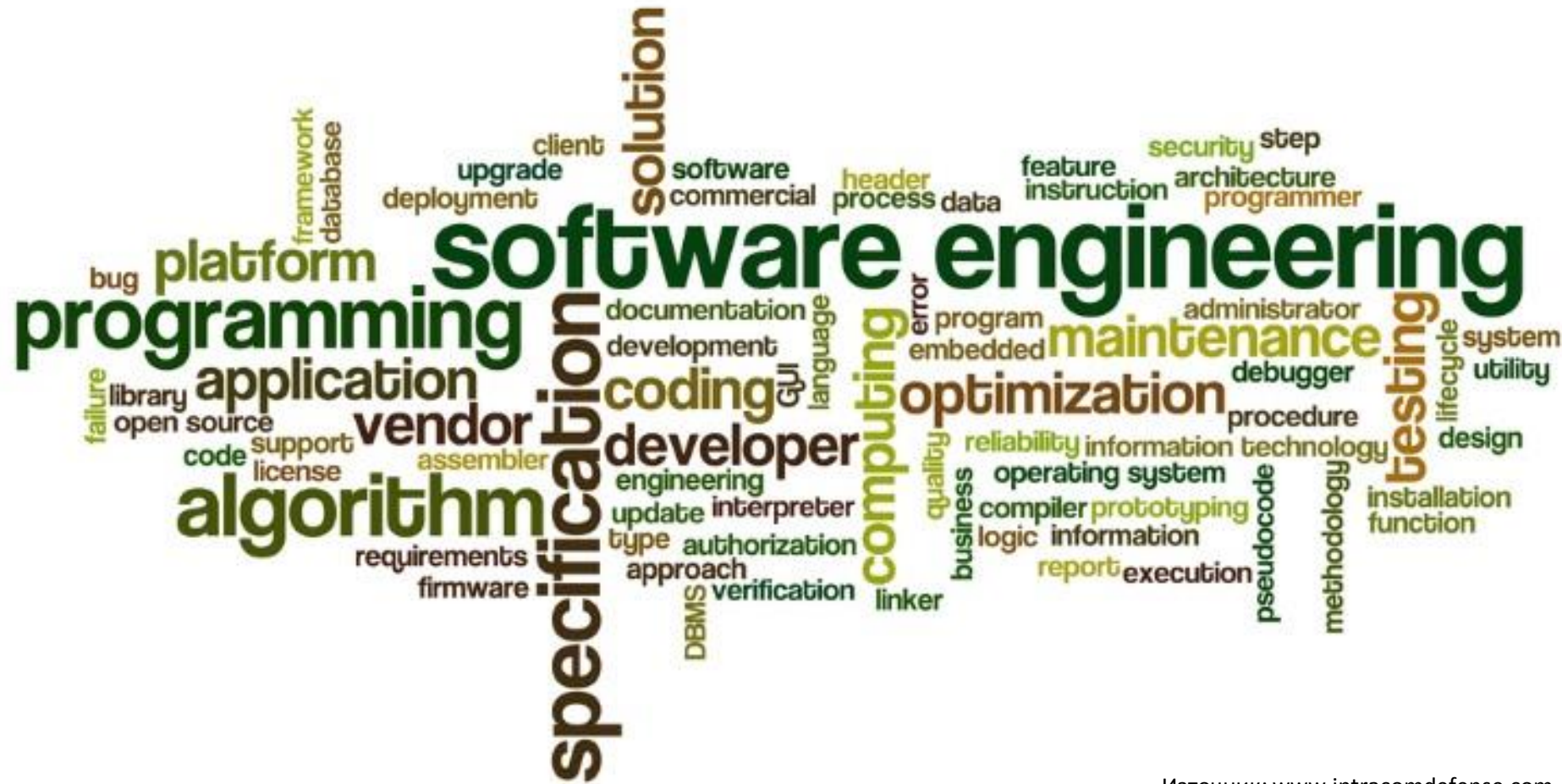
Съдържание

- Софтуерно инженерство (основни понятия)
- Основни понятия от компютърната архитектура
- Бройни системи
- Развитие на езиците за програмиране
- Понятие за софтуер

Какво е софтуерно инженерство?



Софтуерно инженерство

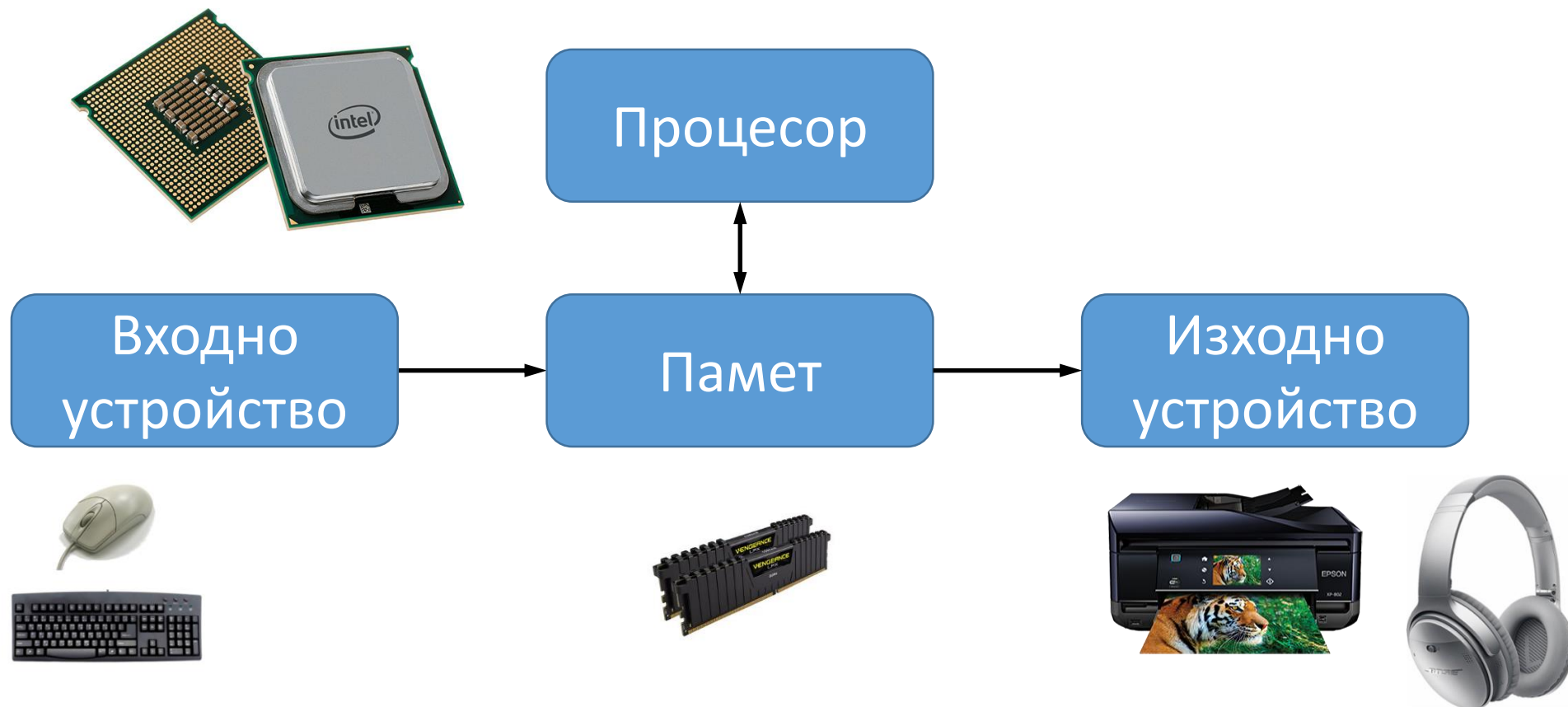


Източник: www.intracomdefense.com

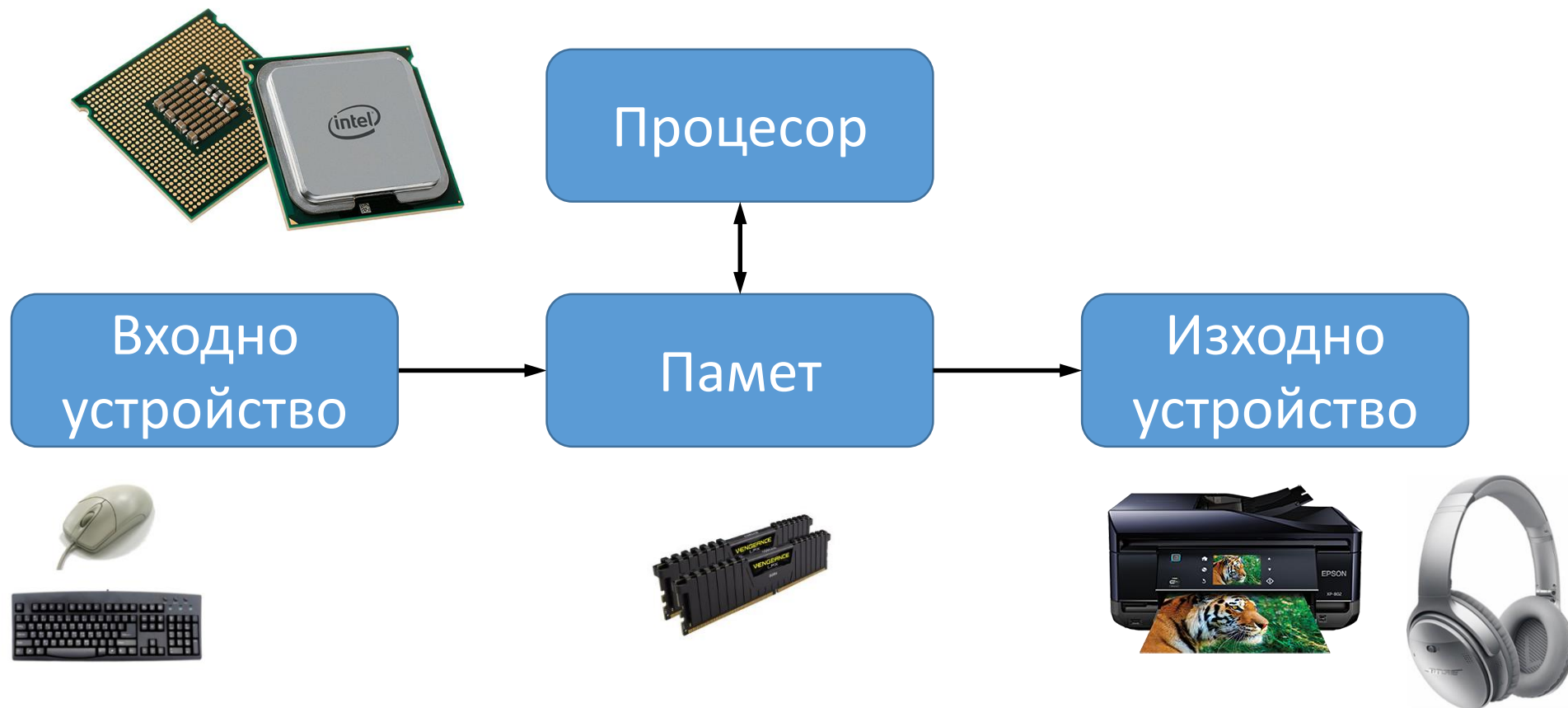
Софтуерно инженерство

- Софтуерното инженерство е дисциплина, целта на която е да се създаде софтуер по предвидим начин и с предвидими характеристики
- Основна дейност в тази дисциплина все още е писането на код, т.е. - програмирането

Какво представлява компютърът



- Най-общо, компютърът е средство за представяне и обработване на информация.
- Компютърът е устройство или система, която може да изпълнява определена последователност от операции
- Това са най-често числови изчисления, аналитични преобразувания или обработка на данни, но включват също и входно/изходни операции.



Памет

- Паметта на компютъра може да бъде изобразена като редица от елементи, всеки от които е носител на информацията – цифрите 0 и 1. Такъв обем информация се нарича **бит (binary digit)**.
- Технически не е възможно да бъде реализиран пряк достъп до всеки бит от паметта, поради което няколко бита се групират в дума. Думата може да е с размер 8, 16, 32, 64 и т.н. бита.
- 8 – битовата информационна единица се нарича **байт**.



Памет ...

- Обемът на паметта се измерва в байтове (B), килобайтове (KB), мегабайтове (MB), гигабайтове (GB) и терабайтове (TB), където:
 - $1 \text{ KB} = 2^{10} \text{ B} = 1024 \text{ B}$
 - $1 \text{ MB} = 2^{10} \text{ KB} = 1024 \text{ KB}$
 - $1 \text{ GB} = 2^{10} \text{ MB} = 1024 \text{ MB}$
 - $1 \text{ TB} = 2^{10} \text{ GB} = 1024 \text{ GB}$

Памет ...

- Обикновено се реализира пряк достъп до всяка дума на паметта. Всяка дума се свързва с пореден номер, наречен неин адрес. Номерацията започва от 0.
- Размерът в битове на думата е свързан с разредността на компютъра (32 бита, 64 бита).

0	...
1	...
	...
999	...

Какво представлява информацията в компютъра

Адрес в паметта	Информация
00000000	10010111
00000001	11100101
00000010	10111100
00000011	10101000
00000100	11111100
00000101	00000000
00000110	11110100
.....
11111110	10100011
11111111	11000100

Бройни системи

- Числа и цифри
- Десетична бройна система
 - $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$
 - $34_{(10)} = 3 \cdot 10^1 + 4 \cdot 10^0$
- Двоична бройна система
 - $\{0, 1\}$
 - $100010 = 1 \cdot 2^5 + 0 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = 34$
- Шестнадесетична бройна система
 - $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F\}$
 - За да се отличават от десетичните, шестнадесетичните числа се обозначават с „0x“ пред запис на числото
 - $0x1F = 1 \cdot 16^1 + 15 \cdot 16^0 = 31$

Бройни системи

Всяка позиция в записа на числото се нарича *разряд*

Основата указва в коя бройна система е този запис.

198723 (10)

Старши разряд

Младши разряд

Полиномиален и цифров запис

Цифров запис: $a_n a_{n-1} \dots a_1 a_0_{(b)}$

Полиномиален запис: $a_n b^n + a_{n-1} b^{n-1} + \dots + a_1 b + a_0$

$$8023_{(10)} \leftrightarrow 8 \times 10^3 + 0 \times 10^2 + 2 \times 10^1 + 3$$

Шестнадесетична бройна система

0 0000

1 0001

2 0010

3 0011

4 0100

5 0101

6 0110

7 0111

8 1000

9 1001

A 1010

B 1011

C 1100

D 1101

E 1110

F 1111

Степени на числото две

$$\begin{array}{l} 2^1 = 2 \\ 2^2 = 4 \\ 2^3 = 8 \\ 2^4 = 16 \\ 2^5 = 32 \\ 2^6 = 128 \\ 2^7 = 128 \\ 2^8 = 256 \\ 2^9 = 512 \\ 2^{10} = 1\,024 \end{array}$$

Софтуер и хардуер

- На български: програмно осигуряване и апаратно осигуряване
- Най-общо казано за работата на компютъра е необходим следният софтуер:
 - Операционни системи;
 - Приложни програми
- За създаването на приложите програми са необходими и т.нар. интегрирани среди за разработка на софтуер
 - Integrated Development Environment (IDE)

Операционни системи (ОС)

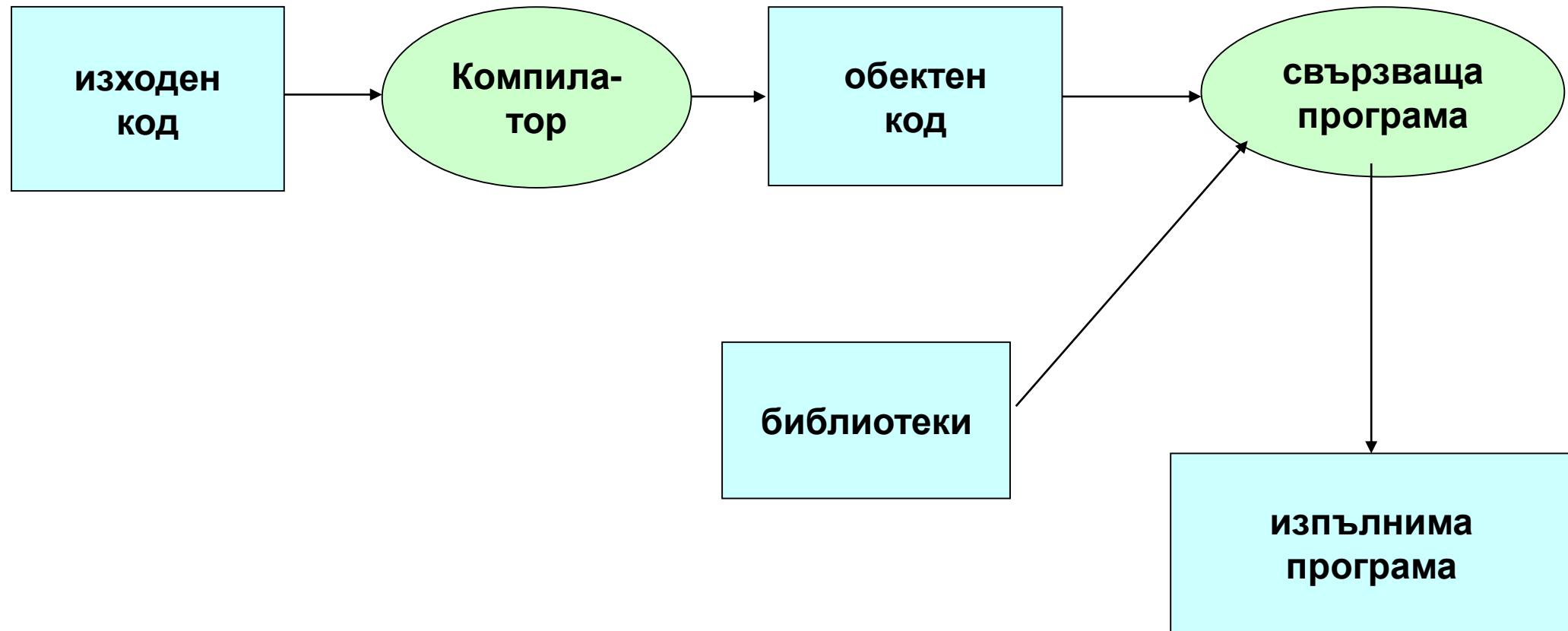
- Всеки компютър работи под управлението на някаква ОС. Тя управлява:
 - взаимодействието между потребителите и хардуера (чрез команден език);
 - периферните устройства (чрез драйвери);
 - файловете (чрез файловата система).
- Широко разпространени ОС са MS DOS, MS WINDOWS, UNIX, LINUX.

Интегрирани среди за разработка

- Какви дейности включва създаването на една компютърна програма
- Но освен това:
 - Проектиране
 - Автоматично създаване на често употребяван код
 - Отстраняване на грешки (debugging)



Примерен процес на създаване на програма

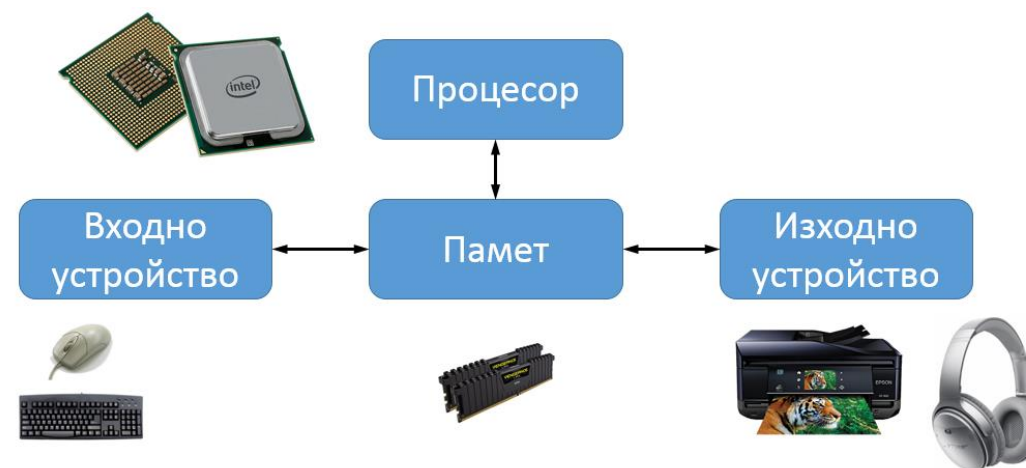


Интегрирани среди за разработка

- Целта им е да улеснят процеса на програмиране и да автоматизират някои дейности от него
- Интегрираните среди за разработка включват:
 - Език за програмиране,
 - Транслатор (компилятор или интерпретатор), свързващ редактор,
 - Изпълнителна система,
 - Система за проверка на програми,
 - Система за поддържане на библиотеки и
 - Текстови редактори.
- Популярни интегрирани среди за разработка
 - Visual Studio
 - Eclipse
 - Code Blocks

Алгоритъм

- Крайна последователност от действия които водят до решаване на даден проблем, свързан с:
 - Изчисления
 - Обработка на данни



Представяне на алгоритъм

- Псевдокод
- Блок-схема
- Програмен език

Псевдокод

```
set total to zero

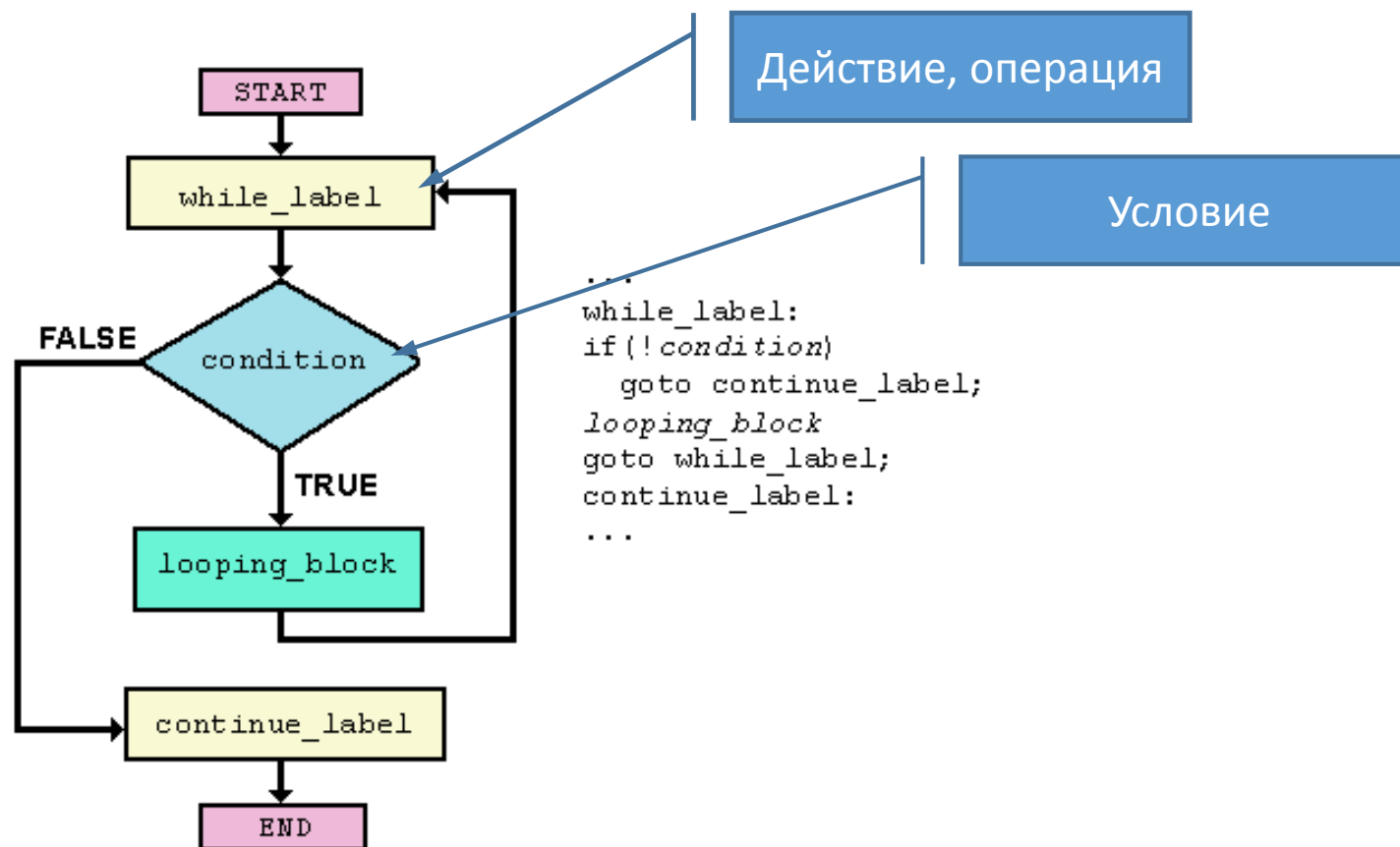
get list of numbers

loop through each number in the list
  add each number to total
end loop

if number more than zero
  print "it's positive" message
else
  print "it's zero or less" message
end if
```

Източник: <https://www.lynda.com/Programming-Foundations-tutorials/Writing-pseudocode/83603/90472-4.html>

Блок схеми



Източник: <http://www.yolkfolk.com/dizzyage/books/gs9/page07.html>

Програмен език (C++)

```
#include <iostream>
```

```
using namespace std;
```

```
int main () {  
    int a = 5;  
    cout << "a = " << a << endl ;  
    cout << "2a = " << 2 * a << endl ;  
    return 0;  
}
```

Развитие на езиците за програмиране

- Машинни езици
- Асемблерни езици
- Процедурно-ориентирани езици
- Обектно-ориентирани езици
- Други:
 - Макроезици
 - Декларативни езици
 - И т.н.

Машинен език

- Всеки процесор си има свой език за програмиране, наречен машинен език и може да изпълнява програми, написани на него.
- Машинният език представлява поредица от двоични числа, които представят инструкциите към процесора
 - Например: 161 15000 45 10 127 25000
- Всяка проста на пръв поглед операция (например събиране на две числа) е съставена от голямо количество подобни машинни инструкции, които са различни за всеки процесор.

Асемблерен език

- Първите стъпки за автоматизиране на процеса на програмиране
- Асемблерният език е общ за цяла фамилия процесори
- Асемблерните инструкции дават кратки имена на цяла поредица от машинни инструкции:
 - ADD – събери
 - MOV – премести
 - CMP al, dl – сравни
 - SUB – извади,
 - JG – ако е по-голямо от 0, премини към

Асемблерен език

- Като се използват тези команди, „програмата“

```
161 15000 45 10 127 25000
```

Изглежда така

```
MOV AX, [15000]
```

```
SUB AX, 10
```

```
JG 25000
```

- Този фрагмент е по-лесен за четене от хора, но за да бъдат разбрани от процесора, те трябва да бъдат преведени на машинен език.
- Тази задача се изпълнява от програма, наречена асемблер.

Езици за програмиране от високо ниво

- Тези езици са още по-подходящи за използване от хора
- За сметка на това при тях е необходима програма от още по-високо ниво, за да преведе кода на машинен език
- Най-често тази програма се нарича компилатор
- Ако се върнем към нашият пример

161 15000 45 10 127 25000

Може да се напише на C++ по следния начин:

```
if (a > 10) cout<<"error";
```

Езици от високо ниво

- Процедурно-ориентирани езици
 - Улесняват многократната употреба на кратки програмни фрагменти под формата на т.нар. процедури или още функции
- Обектно-ориентирани езици
 - Улесняват многократната употреба на код, под формата на т.нар. обекти
- Езици от още по-високо ниво

- За подготовката на тази презентация са използвани слайдове създадени от:
 - Доц. Александър Григоров
 - Доц. Атанас Семерджиев
 - Доц. Трифон Трифонов