

# ПРОГРАМА НА С

Програма на С – данни + алгоритъм за обработката им

Структура на програма на С – набор от равнопоставени програмни единици – функции. С една от тях (int main() ) започва изпълнението на програмата (задава входна точка ). Всяка функция представлява независима програмна единица с изчисляван резултат, име и параметри (входни данни за функцията). Всяка една функция може да извиква всяка друга функция в програмата, като след завършването управлението се връща в извикващата функция. Функциите не могат да се включват една в друга. Завършването на функцията служила като входна точка завършва програмата

- \* Команди на пред процесора
- \* Дефиниране на типове
- \* Прототипи на функции – декларира типа на функцията и параметрите ѝ
- \* Променливи
- \* Функции -> \* заглавен ред (тип, име, формални параметри)
  - \* променливи
  - \* изпълними инструкции

```
(0)      #include <stdio.h>
(1)      typedef ...
(2)      int chng1(int, int);
(3)      int chng2(int,int);
(4)      int chng1(int p, int delta){
(5)          return p+delta;      /*addition */
(6)      }
(7)      int main() {
(8)          int a =5,b;
(9)          a = chng1(a,3);      // call first function
(10)         b = chng2(a,1);
(11)         return a;
(12)     }
(13)     int chng2(int k,int delta){
(14)         return k-delta;      /*subtraction */
(15)     }
```

Ред на изпълнение (8) -> (9)->(5)->(9) ->(10) ->(14) ->(10)-> (11) -> край на програмата.

Езикът прави разлика между главни малки букви (case sensitive) – W и w са 2 различни букви

## ПРЕДСТАВЯНЕ НА ДАННИ В ЕЗИКА С

### Характеристики на данните

**тип** - възможни стойности, начин на представяне в паметта, възможни операции

**клас памет** – цикъл на живот (възникване, начална стойност, разрушаване),видимост

**възможност за промяна на стойността** – константи и променливи. Всяка променлива преди да бъде използвана, трябва да бъде дефинирана с типа си. Константите могат да бъдат задавани със стойността си(литерали) или да бъдат дефинирани с тип и име.

Тип -> **void**(тип използван за специални цели)

**скаларен** -> аритметични, изброим, указатели

**съставен** -> масиви, структури, обединения

**аритметични типове**

<u>char</u>	<u>short</u>	<u>int</u>	<u>long</u>	<u>float</u>	<u>double</u>	unsigned	описание
интегрален				реален			

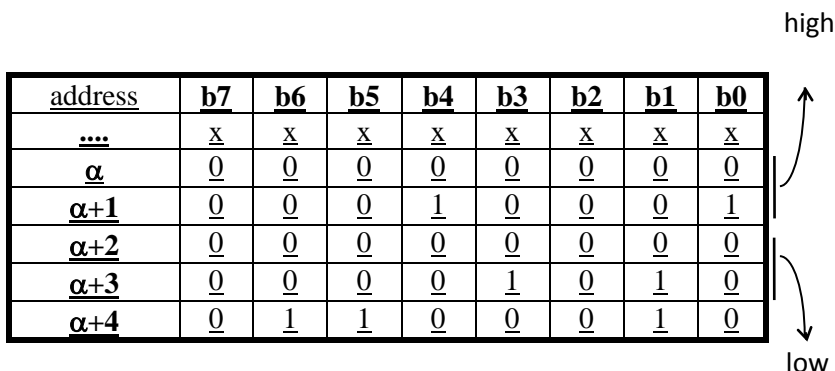
**Променливи.** Всяка променлива има **име**(идентификатор) и стойност. Всеки идентификатор се състои само от букви и цифри и започва задължително с буква.

**ключови думи, разделители**

За да може да се използва променливата, тя трябва да се дефинира (да се създаде с тип)

```
int high,low;
int high=17,low=10;
char c='b'; ( '\0142' 9810)
```

представяне на данни от интегрален тип



[ASCII](#) 0-31; 32-126;

```
#include <stdio.h>
#include <limits.h>
int main() {
    printf("Number of bits in a char object %d\n",CHAR_BIT);
    printf("Minimum value for a signed char %d\n",SCHAR_MIN);
    printf("Maximum value for a signed char %d\n",SCHAR_MAX);
    printf("Maximum value for a unsigned char %d\n",UCHAR_MAX);
    printf("Minimum value for a char %d\n",CHAR_MIN);
    printf("Maximum value for a char %d\n",CHAR_MAX);
    printf("Maximum value for a short %d\n",SHRT_MAX);
    printf("Maximum value for an int %d\n",INT_MAX);
    printf("Minimum value for an int %d\n",INT_MIN);
    printf("Maximum value for an unsigned int %lu\n",UINT_MAX);
}
```

- Number of bits in a char object 8**
- Minimum value for a signed char -128**
- Maximum value for a signed char 127**
- Maximum value for a unsigned char 255**
- Minimum value for a char -128**
- Maximum value for a char 127**
- Maximum value for a short 32767**
- Maximum value for an int 2147483647**
- Minimum value for an int -2147483648**
- Maximum value for an unsigned int 4294967295**

**Константи**    литерали    десетични 26 -37 3.14 12e2 12e2F 345L  
осмични (започващи с 0) 015 (13<sub>10</sub> 1101<sub>2</sub>) 021(19<sub>10</sub> 10001<sub>2</sub>)  
шестнадесетични (започващи с 0x) 0x15 (21<sub>10</sub> 10101<sub>2</sub>)  
символи (в апострофи) 'a' 'z',  
специални символи '\n' '\t' '\\' '\'" '\0' '\044' (00100100)  
низ "this is a string"  
**внимание!** а 'a' "a" a()  
с име и тип    const int k =25;Представяне в паметта на данни от реален тип

Максимални и минимални стойности реални

```
#include <stdio.h>
#include <float.h>
int main() {
    printf("Minimum value of a float: %g\n", FLT_MIN );
    printf("Maximum value of a float: %g\n", FLT_MAX );
    printf("Precision of a float: %d digits\n", FLT_DIG );
    printf("Minimum value of a double: %g\n", DBL_MIN );
    printf("Maximum value of a double: %g\n", DBL_MAX );
    printf("Precision of a double: %d digits\n", DBL_DIG );
}
```

**Minimum value of a float: 1.17549e-038**

**Maximum value of a float: 3.40282e+038**

**Precision of a float: 6 digits**

**Minimum value of a double: 2.22507e-308**

**Maximum value of a double: 1.79769e+308**

**Precision of a double: 15 digits**