

Типове и променливи

гл. ас. д-р. Нора Ангелова

Памет

- Редица от елементи със стойност 0 и 1, наречени битове.
- Клетка от паметта (дума) – групиране на няколко бита.

Памет

Измерва се в:

- bit
- байт (В) – 8bit
- килобайт (КВ) – 2^{10} В ~1000В
- мегабайт (МВ) – 2^{20} В
- гигабайт (ГВ) – 2^{30} В
- терабайт (ТВ) – 2^{40} В

Памет

- Да си заделим памет

място → адрес

име → идентификатор

големина → тип

Идентификатор

- Последователност от букви, цифри и `_`.
- Не може да започва с цифра.
- Не може да се използват ключовите думи в езика.

Пример:

```
int index;  
int nextMonthIndex;  
char firstCharacter;
```

- Езикът различава главните от малките латински букви.

Пример:

```
int index; и int Index; // различни променливи
```

- Показват значението на съответната променлива/функция.

Идентификатор

- Конвенции за именуване:

`int nextIndex; // camel case`

`int NextIndex; // pascal case`

`int next_index; // snake case`

Типове

- Скаларни

`int` – чели числа;

`double` – реални числа;

`float` – реални числа;

`char` – символ;

`bool` – булева стойност (true/false);

`enum` – изброен;

`void`

Указател и псевдоним;

- Съставни – масив, низ, вектор.

Типове

* Стойностите могат да бъдат различни в зависимост от компютъра, компилатора и др.

- `char` – 1B
- `int` – 4B $\in [-2,147,483,648 \text{ to } 2,147,483,647]$
- `unsigned int` $\in [0 \text{ to } 4,294,967,295]$

$$4\text{B} = 32 \text{ bit} \rightarrow 2^{32} = 4,294,967,295$$

- `float` – 4B (3.4E +/- 38 - **7 digits**)
- `double` – 8B (1.7E +/- 308 - **15 digits**)

Да проверим размера на даден тип

```
#include <iostream>
using namespace std;

int main() {
    cout << "Size of char: " << sizeof(char) << endl;

    cout << "Size of int: " << sizeof(int) << endl;

    cout << "Size of float: " << sizeof(float) << endl;

    cout << "Size of double: " << sizeof(double) << endl;

    return 0;
}
```

Преобразуване на типове

```
double x = 2.3;
```

- Неявно преобразуване

```
int y = x; // 2
```

- (тип)<израз>

```
(int)(1.52 + 56.2) // 57
```

```
(double)(123 + 18) // 141.0
```

- static_cast<тип>(израз)

```
static_cast<int>(1.52 + 56.2) // 57
```

```
static_cast<double>(123 + 18) // 141.0
```

Символен тип

- Състои се от крайно и наредено множество от символи

ASCII

```
(int)'F';    // 70  
(double)65; // A
```

- Символите могат да бъдат сравнявани

```
'F' < 'Z'    // true
```

Декларация на променлива

- `<тип> <име_на_променлива>{, <име_на_променлива>}опц;`

`<тип>` - тип данни;

`<име_на_променлива>` - идентификатор;

Пример:

```
int index;
```

```
int nextMonthIndex;
```

ИЛИ

```
int index, nextMonthIndex;
```

Инициализация на променлива

- $\langle \text{тип} \rangle \langle \text{име_на_променлива} \rangle = \langle \text{стойност} \rangle$
 $\{, \langle \text{име_на_променлива} \rangle = \langle \text{стойност} \rangle \}_{\text{опц}};$

```
int nextIndex = 1;
```

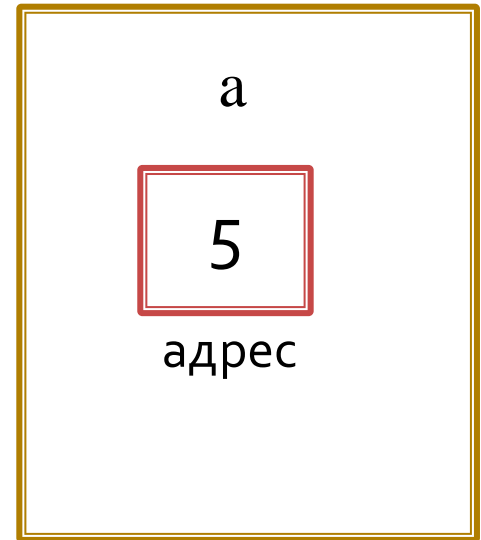
- $\langle \text{тип} \rangle \langle \text{име_на_променлива} \rangle (\langle \text{стойност} \rangle)$
 $\{, \langle \text{име_на_променлива} \rangle (\langle \text{стойност} \rangle) \}_{\text{опц}};$

```
int nextIndex(1);
```

Променливи

- Адрес
- Присвояване на стойност

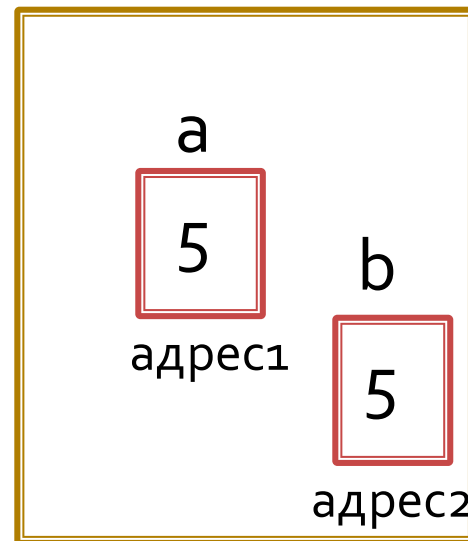
```
int a = 5;
```



Променливи

- Адрес
- Присвояване на стойност

```
int a = 5;  
int b = a;
```



Област на променливите

- Започва от нейната дефиниция.
- Продължава до края на блока (оператора).

...

```
double a;
```

...

```
} – края на блока
```


Оператор за вход и изход

- `cin >> <променлива> {>> <променлива>}опц;`

Използва се за въвеждане на стойност на променлива от потребителя.

- `cout << <променлива> {<< <променлива>}опц;`

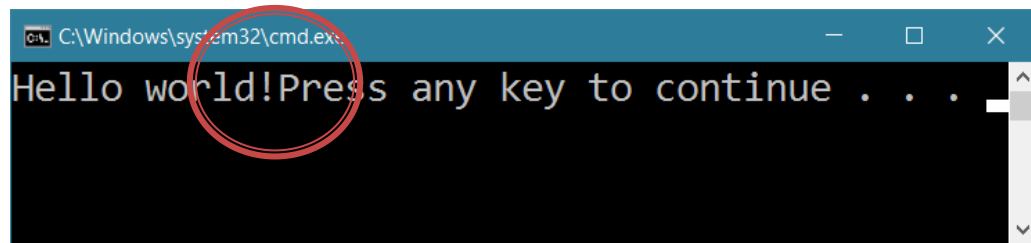
Извежда стойността на променливата.

Първата ни програма

```
#include <iostream>
using namespace std;

int main () {
    // Извежда "Hello world!"
    cout << "Hello world!";

    return 0;
}
```



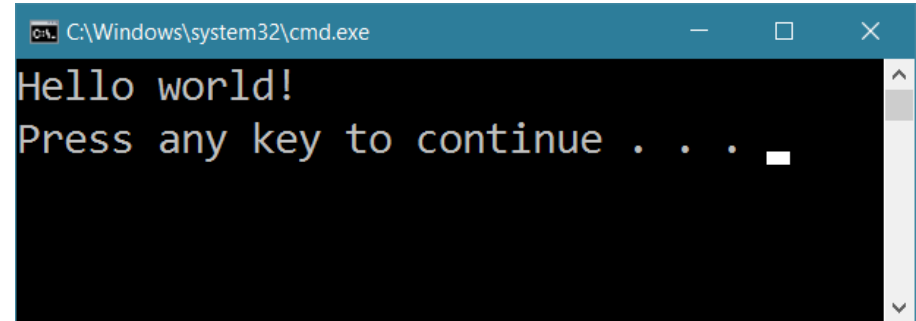
A screenshot of a Windows command prompt window. The title bar shows the path "C:\Windows\system32\cmd.exe". The window content displays the text "Hello world!Press any key to continue . . ." in a monospaced font. A red circle is drawn around the word "Press" in the output text.

Първата ни програма

```
#include <iostream>
using namespace std;

int main () {
    // Извежда "Hello world!"
    cout << "Hello world!\n";

    return 0;
}
```



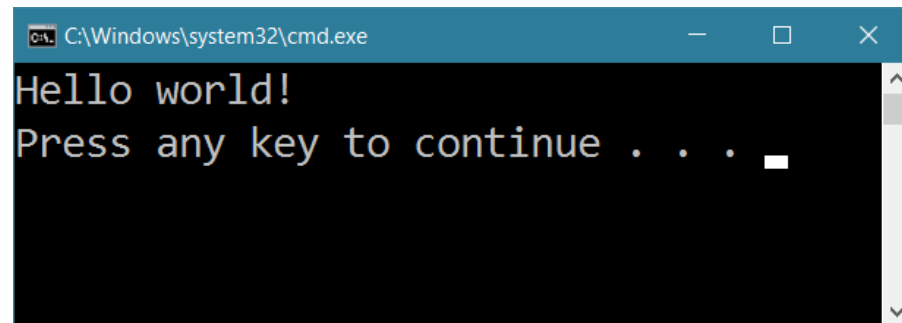
A screenshot of a Windows command prompt window. The title bar shows the path "C:\Windows\system32\cmd.exe". The window content displays the output of a program: "Hello world!" followed by "Press any key to continue . . .". A small white cursor is visible at the end of the second line.

Първата ни програма

```
#include <iostream>
using namespace std;

int main () {
    // Извежда "Hello world!"
    cout << "Hello world!" << endl;

    return 0;
}
```



A screenshot of a Windows command prompt window. The title bar shows the path "C:\Windows\system32\cmd.exe". The window content displays the output of a C++ program: "Hello world!" followed by "Press any key to continue . . .". A small white cursor is visible at the end of the second line.

Следва продължение . . .