# Изчислителен процес. Блок схеми. Състояние на програмата. Блокове в С++. Начални сведения за масиви и символни низове

Калин Георгиев

28 октомври 2016 г.

#### Препоръчителна литература

#### Основна

- М. Тодорова. Програмиране на С++ първа част, С. СИЕЛА , 2002.
- М. Тодорова, П. Армянов, Д. Зотева, К. Георгиев. Сборник от задачи по програмиране на С++. Част първа. Увод в програмирането, ТехноЛогика ЕООД, 2008.

#### Допълнителна

- Niklaus Wirth, Algorithms + Data Structures = Programs, Prentice-Hall Series in Automatic Computation
- Robert Sedgewick, Algorithms in C++
- П. Наков, П. Добриков, Програмиране = ++Алгоритми; С., Тор Теат Со, 2003. Допълнителна:
- Л. Амерал, Алгоритми и структури от данни в С++, С, СОФТЕХ, 2001.
- Липман, Езикът С++ в примери, С., КОЛХИДА ТРЕИД КООП, 1993.

Изчислителен процес

#### int a,b;

$$a = 5$$

$$b = 10$$

$$b = a + b$$

$$h = a + b$$

 а	b	
 ?	?	
а	b	
а	b	
5		
а	b	
5		
а	b	
5		

$$a = 5;$$

$$b = 10$$

$$b = a + b$$

$$h = a + b$$

 a	b	
 ?	?	
 а	b	
 5	?	
а	Ь	
5		
а	Ь	
5		
а	Ь	
5		

$$a = 5;$$

$$b = 10;$$

$$b = a + b$$

$$b = a + b$$

 a	b	
 ?	?	
 а	b	
 5	?	
 а	b	
 5	10	
а	b	
5		
а	Ь	



$$a = 5;$$

$$b = 10;$$

$$b = a + b$$
;

$$b = a + b$$

 а	b	
 ?	?	
 а	b	
 5	?	
 а	b	
 5	10	
 а	b	
 5	15	
а	b	
5		

$$a = 5;$$

$$b = 10;$$

$$b = a + b;$$

$$b = a + b;$$

 а	b	
 ?	?	
 а	b	
 5	?	
 а	b	
 5	10	
 а	b	
 5	15	
 а	b	
 5	20	

Блок схеми и процеси



#### Линейна програма

#### Линейна програма

#### Разклонение

#### Разклонение

# Цикъл

# Цикъл



#	Х	у	Z
0	2	5	1
1	2	4	2
2	2	3	4
3	2	2	8
4	2	1	16
5	2	0	32

Още един вид цикъл в С++



#### Неструктурирани езици

```
10. Въведи X и Y
20. Z := 1
30. Aко Y == 0 GOTO 70
40. Z := Z * X
50. Y := Y -1
60. GOTO 30
70. Отпечатай Z
80. Край
```

## Цикъл While

```
int x,y,z;
cin >> x >> y;
z = 1;
while (y != 0)
{
   z = z * x;
   y--;
}
cout << z;</pre>
```

# Сравнение

```
int x,y,z;

cin >> x >> y;

z = 1;

while (y != 0)

z = z * x;
y--;
}

10. Въведи X и Y

20. Z := 1

30. Ako Y == 0 GOTO 70

40. Z := Z * X

50. Y := Y -1

60. GOTO 30

70. Отпечатай Z

80. Край

2 = z * x;
y--;
}
```

#### Примери

- Намиране на броя на цифрите в десетичния запис на естествено число
- n!
- Намиране на сумата на цифрите в десетичния запис на естествено число
- Проверка дали дадено естествено число притежава цифрата 5 в десетичния си запис

Блокове в С++



#### Какво е блок?

```
int a = 0;
int b = a + 10;
if (a == 0)
  int b = a;
 cout << b;
 b = 100;
cout << b;
```

```
int a = 0;
int b = a + 10;
```

 а	b	
 0	10	

```
int b = a;
cout << b;
```

а	b	
	10	

b	100;

а	Ь	b	
5	10		
а	Ь		
	10		
а	Ь		
	10		

cout << b

```
int a = 0;
int b = a + 10;
```

 а	b	
 0	10	

```
if (a == 0)
{
  int b = a;
  cout << b;</pre>
```

 а	b	b	
 0	10	0	

b	100;

а	Ь	b	
5	10		
а	Ь		
	10		
а	b		
	10		

cout << b

```
int a = 0;
int b = a + 10;
```

 а	b	
 0	10	

```
if (a == 0)
{
  int b = a;
  cout << b;</pre>
```

 а	b	b	
 0	10	0	

b	=	100;

 а	b	b	
 5	10	100	

1 +	//	h.			

 а	b	
 0	10	

```
if (a == 0)
{
  int b = a;
  cout << b;</pre>
```

10

cout << b

0

Χ

 а	b	
 0	10	

 а	b	b	
 0	10	0	

 а	b	b	
 5	10	100	
_	h	Y	

cout << b;

10

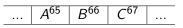
0

Първоначални сведения за масиви и низове



#### Представяне на низове

• ASCii таблица



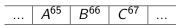
• Представяне в паметта

	Н	Е	L	L	0	
	72	69	76	76	79	

• Масиви

#### Представяне на низове

• ASCii таблица



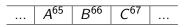
• Представяне в паметта

 Н	Е	L	L	0	_	
 72	69	76	76	79	0	

• Масиви

#### Представяне на низове

• ASCii таблица



• Представяне в паметта

 Н	Е	L	L	0	_	
 72	69	76	76	79	0	

• Масиви

#### Масиви

• Дефиниране чрез тип и размер:

```
int arr[100];
```

• Достъп до всеки отделен елемент:

```
int b = arr[18];
cin >> arr[18];
b = arr[1] + arr[2];
```

• Обхождане с for цикъл

```
for (int count = 0; count < 100; count++
{
   cout << arr[count];
}</pre>
```

#### Масиви

• Дефиниране чрез тип и размер:

```
int arr[100];
```

• Достъп до всеки отделен елемент:

```
int b = arr[18];
cin >> arr[18];
b = arr[1] + arr[2];
```

Обхождане с for цикъл

```
for (int count = 0; count < 100; cou
{
   cout << arr[count];
}</pre>
```

#### Масиви

• Дефиниране чрез тип и размер:

```
int arr[100];
```

• Достъп до всеки отделен елемент:

```
int b = arr[18];
cin >> arr[18];
b = arr[1] + arr[2];
```

• Обхождане с for цикъл

```
for (int count = 0; count < 100; count++)
{
   cout << arr[count];
}</pre>
```

#### Прост пример с низове

```
int main ()
ſ
  char str[100] = "Helloworld!";
  cout << str << endl;
  str[0] = 'Y';
  cout << str << endl;
  cout << "Please input austring:";
  cin >> str;
  for (int counter = 0; counter < 100; counter++)</pre>
    if (str[counter] == 'a')
      str[counter] = 'b';
  cout << str << endl;
  return 0:
```

#### Какво не можем да праивм с масиви и низове

• Няма проверка за коректност!

```
char a[6] = "HELLO";
a[10] = '!';
```

 0	1	2	3	4	5	 10	
 Н	Е	L	L	0	_	 Ţ	

- Присвояване (a=b)
- Сравнение (a==b, a < b,...)

#### Какво не можем да праивм с масиви и низове

• Няма проверка за коректност!

```
char a[6] = "HELLO";
a[10] = '!';
```

 0	1	2	3	4	5	 10	
 Н	Е	L	L	0		 ļ ļ	

- Присвояване (a=b)
- Сравнение (a==b, a < b,...)

#### Какво не можем да праивм с масиви и низове

• Няма проверка за коректност!

 0	1	2	3	4	5	 10	
 Н	Е	L	L	0		 ļ ļ	

- Присвояване (a=b)
- Сравнение (a==b, a < b,...)

## Вградени функции за работа с низове

#### Дължина на низ

```
cout << strlen(a);</pre>
```

#### Присвояване на низове

```
strcpy (d,a);
strcpy (c,a); //!!!
```

#### Сравнение на низове

```
if (strcmp (a,b) < 0)
    {cout << "a<b";}
else
    {cout << "b<=a";};</pre>
```

#### Конкатенация на низове

```
strcpy (d,a); //d -> "HELLO"
```

 0	1	2	3	4	5	6	7	8	9	10	
 Н	E	L	L	0		?	?	?	?	?	

#### strcat (d,b); //d -> "HELLOWORLD"

 .	0	1	2	3	4	5	6	7	8	9	10	
	Н	Е	L	L	0	W	0	R	L	D		

# #include <cstring> ... char a[6] = "HELLO"; char b[6] = "WORLD"; char c[4] = "BYE"; char d[11] = "???";

 0	1	2	3	4	5	
 Н	Е	L	L	0		
 0	1	2	3	4	5	
 W	0	R	L	D		
 0	1	2	3		_	
 В	Υ	E			_	