

# Жизнен цикъл на обектите. Конструктори

Калин Георгиев

26 март 2014 г.

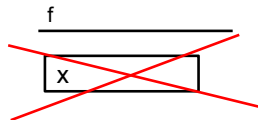
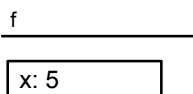
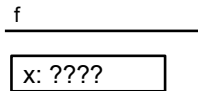
## Жизнен цикъл на обектите (променливите)

## Локална променлива

```
void f ()  
{  
  int x;
```

```
    x = 5;
```

```
}
```



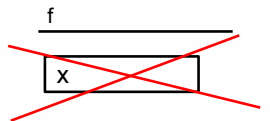
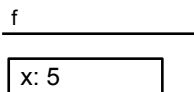
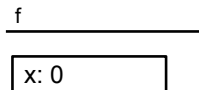
# Параметър

```
void f (int x)
{
  x = 5;
}
```

```
int main ()
{
  f (0);
}
```

x = 5;

```
void f (int x)
{
  x = 5;
}
```

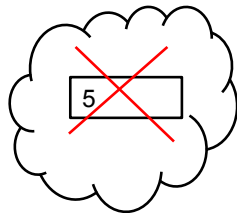
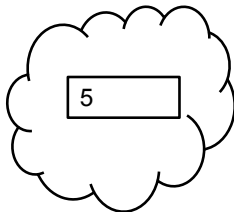
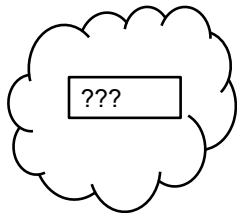


## Динамична памет

```
int* x = new int;
```

```
*x = 5;
```

```
delete x;
```

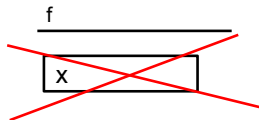
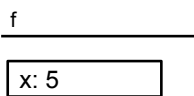
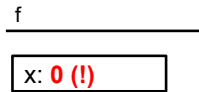


## Инициализация по време дефиниция

## Локална променлива

```
void f ()
{
  int x = 0;           x = 5;           }

```

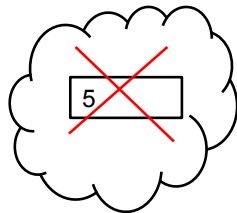
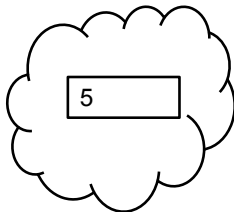
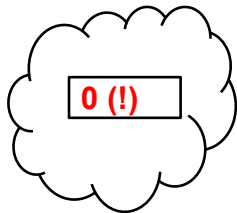


## Динамична памет

```
int* x = new int (0);
```

```
*x = 5;
```

```
delete x;
```





# Конструктори

# С какви стойности е “позволена инициализацията”

- “Стандартни”, “вградени”, “прости” типове

```
double x = 1.2; // <=> double x (1.2);  
double y = 1;  
double z = 'q';
```

# Потребителски дефинирани типове (пример: CharSet)

- Инициализация с низ (char\*)

```
CharSet s1 = "hello";
```

- Инициализация с буква (char)

```
CharSet s2 = 'b';
```

- Инициализация с число

```
CharSet s3 = 5;
```

# Потребителски дефинирани типове (пример: CharSet)

- Инициализация с низ (char\*)

```
CharSet s1 = "hello";
```

- Инициализация с буква (char)

```
CharSet s2 = 'b';
```

- Инициализация с число

```
CharSet s3 = 5;
```

# Потребителски дефинирани типове (пример: CharSet)

- Инициализация с низ (char\*)

```
CharSet s1 = "hello";
```

- Инициализация с буква (char)

```
CharSet s2 = 'b';
```

- Инициализация с число

```
CharSet s3 = 5;
```

# Конструктор (от буква)

```
class CharSet
{
    //...

    void init ()
    { //CharSet* this
        for (int i =0; i < 26; i++)
            this->contents[i] = false;
    }

    //initialize with a single character
    CharSet (char c)
    {
        init ();
        this->contents[c-'a'] = true;
    }
};
```

# Конструктор (от низ)

```
class CharSet
{
    //...

    CharSet (char *s)
    {
        init ();
        for (int i = 0; i < strlen(s); i++)
        {
            this->contents[s[i]-'a'] = true;
        }
    }
};
```

## Конструктор по подразбиране



# Без инициализация

```
CharSet s1;  
  
//s1.init();  
s1.print (); //NOT OK!!!
```

# Задължителна (по подразбиране) инициализация

```
class CharSet
{
    //...
    CharSet ()
    {
        init ();
    }
};
int main ()
{
    CharSet s1;

    //s1.print ();
    s1.print (); //OK!
}
```

## Други “автоматични” конструктори

# “Вградени” конструктори

- Инициализация с друг обект от същия тип

```
CharSet s1 = "hello";  
CharSet s2 = s1;  
  
s2.print();
```

Благодаря за вниманието!