

Видове променливи



Локални променливи

- декларирани **във** функцията
- видими са от мястото на деклариране до **края на функцията или блока** в който се намират
- съществуват докато функцията завърши изпълнението и

Глобални променливи

- декларирани са **ИЗВЪН** всички функции
- видими са от мястото на деклариране до **края на програмата**
- съществуват докато завърши изпълнението на програмата

Пример: видове променливи

```
int a; // глобална променлива
```

```
void MyFunction() {  
    int b; // локална променлива  
    b = 5;  
    a = 7;  
}
```

```
int main() {  
    a = 3;  
    return 0;  
}
```

Кога кое да ползваме?

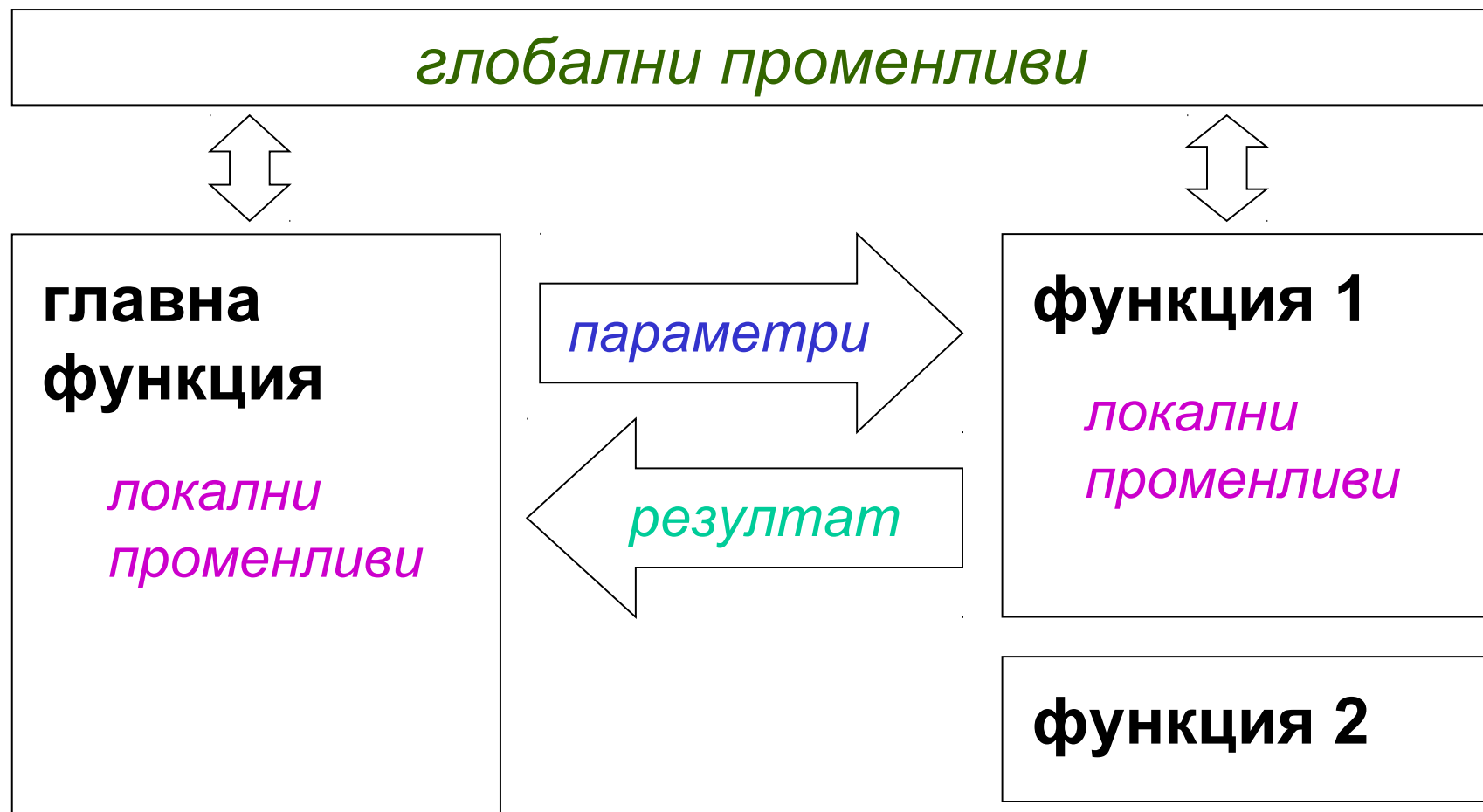
- Функцията е като инструмент – създадена е да върши нещо, и да върне **резултат**
- Като **формални параметри** и подаваме само онова, което е необходимо да знае, за да свърши работата си
- Ако за изчисленията и са нужни допълнителни величини, те се декларираат в нея като **локални променливи**
- **Глобални променливи** декларираме, когато няколко функции ползват общи данни
- *Глобалните променливи трябва да се ползват пестеливо, защото правят функциите зависими една от друга*

Как изглеждат декларациите?

```
глобални променливи;  
тип име(списък параметри)  
{  
    локални променливи;  
    команди;  
}
```

- ▣ **тип** – определя резултата който връща функцията
- ▣ **списък параметри** – описват данните, които трябва да подадем на функцията за да работи
- ▣ **локални променливи** – описват данните, специфични само за тази функция
- ▣ **глобални променливи** – ако се налага няколко функции да ползват общи данни

Движението на данните



Вярно или невярно?

Напишете
функция, която
намира сумата
на две числа

```
int a, b, sum;  
void CalcSum() {  
    sum=a+b;  
}
```


Неудачно!

Напишете
функция, която
намира сумата
на две числа

```
int a, b, sum;  
void CalcSum() {  
    sum=a+b;  
}  
...  
a=5;  
b=4;  
CalcSum();  
cout<<sum<<endl;
```

Вярно или невярно?

Напишете
функция, която
намира сумата
на две числа

```
void CalcSum(int a, int b,  
              int sum) {  
    sum=a+b;  
}  
...  
int sum;  
CalcSum(4, 5, sum);  
cout<<sum<<endl;
```

Невярно!

Напишете
функция, която
намира сумата
на две числа

```
void CalcSum(int a, int b,  
              int sum) {  
    sum=a+b;  
}  
...  
int sum;  
CalcSum(4, 5, sum);  
cout<<sum<<endl;
```

Неефективно!

Напишете
функция, която
намира сумата
на две числа

```
void CalcSum(int a, int b,  
              int &sum) {  
    sum=a+b;  
}  
...  
int sum;  
CalcSum(4, 5, sum);  
cout<<sum<<endl;
```

Вярно!

Напишете
функция, която
намира сумата
на две числа

```
int CalcSum(int a, int b) {  
    int sum;  
    sum=a+b;  
    return sum;  
}  
...  
cout<<CalcSum(4, 5)<<endl;
```

ЕФЕКТИВНО!

Напишете
функция, която
намира сумата
на две числа

```
int CalcSum(int a, int b) {  
    return a+b;  
}  
...  
cout<<CalcSum(4, 5)<<endl;
```

Вярно или невярно?

Напишете
функции, които
намират лицето
и обиколка на
кръг

```
double r;  
const double pi=3.14;  
double CalcLice() {  
    return pi*r*r;  
}  
  
double CalcObikolka() {  
    return 2*pi*r;  
}
```

Неудачно!

Напишете
функции, които
намират лицето
и обиколка на
кръг

```
double r;  
const double pi=3.14;  
double CalcLice() {  
    return pi*r*r;  
}  
  
double CalcObikolka() {  
    return 2*pi*r;  
}
```


Вярно или невярно?

Напишете
функции, които
намират лицето
и обиколка на
кръг

```
double CalcLice(double r) {  
    const double pi=3.14;  
    return pi*r*r;  
}  
  
double CalcObikolka(double r)  
{  
    const double pi=3.14;  
    return 2*pi*r;  
}
```

Неефективно!

Напишете
функции, които
намират лицето
и обиколка на
кръг

```
double CalcLice(double r) {  
    const double pi=3.14;  
    return pi*r*r;  
}  
  
double CalcObikolka(double r)  
{  
    const double pi=3.14;  
    return 2*pi*r;  
}
```

ЕФЕКТИВНО!

Напишете
функции, които
намират лицето
и обиколка на
кръг

```
const double pi=3.14;  
double CalcLice(double r) {  
    return pi*r*r;  
}  
double CalcObikolka(double r) {  
    return 2*pi*r;  
}
```

Край

