

# Семинар №2

## ФАКИ 2016

Бирюков В. А.

November 17, 2016

# Переменные

- В языке C все переменные нужно объявить перед использованием
- При объявлении – выделяется память под переменную
- Области видимости переменной
- Название переменной может содержать латинские буквы, цифры и \_
- Название переменной не может начинаться с цифры

# Объявление переменных

- Переменную нужно объявить перед использованием
- Примеры объявления:  
`int n;`  
`float p;`
- `int` – целочисленный тип
- `float` – тип чисел с плавающей точкой

# Инициализация переменных

- Переменные инициализируются с помощью оператора присваивания =
- Примеры:  
n = 3;  
float p = 5.4;  
int a, b, c = 9;

# Логические операторы

Возвращают тип `bool`

!	не
	или
&&	и

# Приоритет операторов

- 1 `()`, `[]`
- 2 `++`, `--`, `+`, `-` (унарные), `sizeof`
- 3 `*`, `/`, `%`
- 4 `+`, `-`
- 5 `>`, `<`, `<=`, `>=`
- 6 `==`, `!=`
- 7 `&`, `|`, `&&`, `||`
- 8 `=`, `+=`, и т.д.

Приоритет операторов C подробнее:

[ru.cppreference.com/w/c/language/operator\\_precedence](http://ru.cppreference.com/w/c/language/operator_precedence)

# Управляющие конструкции

# Базовые управляющие конструкции

if, if else

```
if ( x < 5 )  
    printf( "%d\n" , x );
```

```
if (x < 5) {  
    x++;  
    printf( "%d\n" , x );  
}
```



# Базовые управляющие конструкции

if, if else

```
if ( x < 5 ) {  
    printf("Less than 5\n");  
}  
else if ( x == 5 ) {  
    printf("Equal to 5\n");  
}  
else {  
    printf("More than 5\n");  
}
```

# Базовые управляющие конструкции

Тернарный оператор :?

expression ? expression : expression

$z = (x > 0) ? x : -x$

$\min = (x < y) ? x : y$

# Базовые управляющие конструкции

## Цикл while

```
int i = 0;

while (i < 3)
{
    i++;
    printf("%d ", i);
}
```

Напечатает 1 2 3

# Базовые управляющие конструкции

## Цикл do while

```
int i = 0;

do
{
    i++;
    printf( "%d ", i );
} while ( i < 3 );
```

Напечатает 1 2 3

# Базовые управляющие конструкции

## Цикл for

```
for (int i = 0; i < 3; ++i) {  
    printf("%d ", i);  
}
```

Напечатает 1 2 3

# Базовые управляющие конструкции

## Замечания

В условии циклов может стоять любой оператор:

```
int i = 0;
```

```
while (i++ < 3)  
    printf("%d ", i);
```

```
for (; ; i++) {  
    printf("%d\n", i);  
}
```

# Управляющие конструкции

## Оператор break

```
for (int i = 0; i < 10; ++i) {  
    if (i == 6) {  
        break;  
    }  
    printf ("%d ", i);  
}
```

# Управляющие конструкции

## Оператор continue

```
for (int i = 0; i < 10; ++i) {  
    if (i == 6) {  
        continue;  
    }  
    printf ("%d ", i);  
}
```



# Управляющие конструкции

## Оператор выбора switch

```
switch (i) {  
    case 1:  
        printf("It 's one!\n");  
        break;  
    case 2:  
        printf("It 's two!\n");  
        break;  
    default:  
        printf("It 's something else!\n")  
}
```

# Управляющие конструкции

## Оператор безусловного перехода goto

- Оператор goto передает управление на оператор, помеченный меткой
- Оператор goto в языках высокого уровня является объектом критики, поскольку чрезмерное его применение приводит к созданию нечитаемого кода
- Использование goto в практике программирования на языке C настоятельно не рекомендуется

# Литература

- ❶ Брайан У. Керниган, Деннис М. Ритчи "Язык программирования С"
- ❷ Томас Кормен, Чарльз Лейзерстон, Рональд Ривест, Клиффорд Штайн "Алгоритмы: построение и анализ"