

# Вопросы по лекции 5

Артамонов Ю.Н.

Международный университет  
природы, общества и человека "Дубна"  
филиал Котельники

14 февраля 2018 г.

# Вопрос 1

Дайте ответ на следующие вопросы.

- Вопрос: модуль программы на С называется:

# Вопрос 1

Дайте ответ на следующие вопросы.

- Вопрос: модуль программы на С называется:
- Ответ: функцией.

# Вопрос 1

Дайте ответ на следующие вопросы.

- Вопрос: модуль программы на С называется:
- Ответ: функцией.
- Вопрос: обращение к функции осуществляется посредством:

# Вопрос 1

Дайте ответ на следующие вопросы.

- Вопрос: модуль программы на С называется:
- Ответ: функцией.
- Вопрос: обращение к функции осуществляется посредством:
- Ответ: вызова функции.

# Вопрос 1

Дайте ответ на следующие вопросы.

- Вопрос: модуль программы на С называется:
- Ответ: функцией.
- Вопрос: обращение к функции осуществляется посредством:
- Ответ: вызова функции.
- Вопрос: переменная, которая известна только внутри функции, в которой она определена, называется:

# Вопрос 1

Дайте ответ на следующие вопросы.

- Вопрос: модуль программы на С называется:
- Ответ: функцией.
- Вопрос: обращение к функции осуществляется посредством:
- Ответ: вызова функции.
- Вопрос: переменная, которая известна только внутри функции, в которой она определена, называется:
- Ответ: локальной переменной.

# Вопрос 1

Дайте ответ на следующие вопросы.

- Вопрос: модуль программы на С называется:
- Ответ: функцией.
- Вопрос: обращение к функции осуществляется посредством:
- Ответ: вызова функции.
- Вопрос: переменная, которая известна только внутри функции, в которой она определена, называется:
- Ответ: локальной переменной.
- Вопрос: оператор, который используется в вызываемой функции для передачи значения вызывающей функции называется:



# Вопрос 1

Дайте ответ на следующие вопросы.

- Вопрос: модуль программы на С называется:
- Ответ: функцией.
- Вопрос: обращение к функции осуществляется посредством:
- Ответ: вызова функции.
- Вопрос: переменная, которая известна только внутри функции, в которой она определена, называется:
- Ответ: локальной переменной.
- Вопрос: оператор, который используется в вызываемой функции для передачи значения вызывающей функции называется:
- Ответ: **return**.

# Вопрос 1

Дайте ответ на следующие вопросы.

- Вопрос: модуль программы на С называется:
- Ответ: функцией.
- Вопрос: обращение к функции осуществляется посредством:
- Ответ: вызова функции.
- Вопрос: переменная, которая известна только внутри функции, в которой она определена, называется:
- Ответ: локальной переменной.
- Вопрос: оператор, который используется в вызываемой функции для передачи значения вызывающей функции называется:
- Ответ: **return**.
- Вопрос: ключевое слово, которое используется в заголовке функции, чтобы показать, что функция не возвращает значения или не содержит никаких параметров называется:

# Вопрос 1

Дайте ответ на следующие вопросы.

- Вопрос: модуль программы на С называется:
- Ответ: функцией.
- Вопрос: обращение к функции осуществляется посредством:
- Ответ: вызова функции.
- Вопрос: переменная, которая известна только внутри функции, в которой она определена, называется:
- Ответ: локальной переменной.
- Вопрос: оператор, который используется в вызываемой функции для передачи значения вызывающей функции называется:
- Ответ: **return**.
- Вопрос: ключевое слово, которое используется в заголовке функции, чтобы показать, что функция не возвращает значения или не содержит никаких параметров называется:
- Ответ: **void**.

## Вопрос 2

Дайте ответ на следующие вопросы.

- Вопрос: как называется часть программы, в которой идентификатор может быть использован:

## Вопрос 2

Дайте ответ на следующие вопросы.

- Вопрос: как называется часть программы, в которой идентификатор может быть использован:
- Ответ: область действия.

# Вопрос 2

Дайте ответ на следующие вопросы.

- Вопрос: как называется часть программы, в которой идентификатор может быть использован:
- Ответ: область действия.
- Вопрос: назовите три способа возврата из функции:

# Вопрос 2

Дайте ответ на следующие вопросы.

- Вопрос: как называется часть программы, в которой идентификатор может быть использован:
- Ответ: область действия.
- Вопрос: назовите три способа возврата из функции:
- Ответ: 1. **return**; 2. **return** выражение; 3. достижение правой фигурной скобки, завершающей функцию.

# Вопрос 2

Дайте ответ на следующие вопросы.

- Вопрос: как называется часть программы, в которой идентификатор может быть использован:
- Ответ: область действия.
- Вопрос: назовите три способа возврата из функции:
- Ответ: 1. **return**; 2. **return** выражение; 3. достижение правой фигурной скобки, завершающей функцию.
- Вопрос: какой механизм позволяет компилятору проверять количество аргументов, типы аргументов, порядок их следования, а также тип возвращаемого значения функции:



# Вопрос 2

Дайте ответ на следующие вопросы.

- Вопрос: как называется часть программы, в которой идентификатор может быть использован:
- Ответ: область действия.
- Вопрос: назовите три способа возврата из функции:
- Ответ: 1. **return**; 2. **return** выражение; 3. достижение правой фигурной скобки, завершающей функцию.
- Вопрос: какой механизм позволяет компилятору проверять количество аргументов, типы аргументов, порядок их следования, а также тип возвращаемого значения функции:
- Ответ: прототип функции.

# Вопрос 2

Дайте ответ на следующие вопросы.

- Вопрос: как называется часть программы, в которой идентификатор может быть использован:
- Ответ: область действия.
- Вопрос: назовите три способа возврата из функции:
- Ответ: 1. **return**; 2. **return** выражение; 3. достижение правой фигурной скобки, завершающей функцию.
- Вопрос: какой механизм позволяет компилятору проверять количество аргументов, типы аргументов, порядок их следования, а также тип возвращаемого значения функции:
- Ответ: прототип функции.
- Вопрос: какая функция используется для генерации случайных чисел:

# Вопрос 2

Дайте ответ на следующие вопросы.

- Вопрос: как называется часть программы, в которой идентификатор может быть использован:
- Ответ: область действия.
- Вопрос: назовите три способа возврата из функции:
- Ответ: 1. **return**; 2. **return** выражение; 3. достижение правой фигурной скобки, завершающей функцию.
- Вопрос: какой механизм позволяет компилятору проверять количество аргументов, типы аргументов, порядок их следования, а также тип возвращаемого значения функции:
- Ответ: прототип функции.
- Вопрос: какая функция используется для генерации случайных чисел:
- Ответ: **rand**.

# Вопрос 2

Дайте ответ на следующие вопросы.

- Вопрос: как называется часть программы, в которой идентификатор может быть использован:
- Ответ: область действия.
- Вопрос: назовите три способа возврата из функции:
- Ответ: 1. **return**; 2. **return** выражение; 3. достижение правой фигурной скобки, завершающей функцию.
- Вопрос: какой механизм позволяет компилятору проверять количество аргументов, типы аргументов, порядок их следования, а также тип возвращаемого значения функции:
- Ответ: прототип функции.
- Вопрос: какая функция используется для генерации случайных чисел:
- Ответ: **rand**.
- Вопрос: какая функция выбирает последовательность генератора случайных чисел:

# Вопрос 2

Дайте ответ на следующие вопросы.

- Вопрос: как называется часть программы, в которой идентификатор может быть использован:
- Ответ: область действия.
- Вопрос: назовите три способа возврата из функции:
- Ответ: 1. **return**; 2. **return** выражение; 3. достижение правой фигурной скобки, завершающей функцию.
- Вопрос: какой механизм позволяет компилятору проверять количество аргументов, типы аргументов, порядок их следования, а также тип возвращаемого значения функции:
- Ответ: прототип функции.
- Вопрос: какая функция используется для генерации случайных чисел:
- Ответ: **rand**.
- Вопрос: какая функция выбирает последовательность генератора случайных чисел:
- Ответ: **srand**.

# Вопрос 3

Дайте ответ на следующие вопросы.

- Вопрос: назовите 4 спецификатора класса памяти:

# Вопрос 3

Дайте ответ на следующие вопросы.

- Вопрос: назовите 4 спецификатора класса памяти:
- Ответ: **auto**, **register**, **extern**, **static**.

# Вопрос 3

Дайте ответ на следующие вопросы.

- Вопрос: назовите 4 спецификатора класса памяти:
- Ответ: **auto**, **register**, **extern**, **static**.
- Вопрос: к какому спецификатору класса памяти относится переменная, объявленная в блоке или списке параметров функции, если нет никаких специальных указаний:



# Вопрос 3

Дайте ответ на следующие вопросы.

- Вопрос: назовите 4 спецификатора класса памяти:
- Ответ: **auto**, **register**, **extern**, **static**.
- Вопрос: к какому спецификатору класса памяти относится переменная, объявленная в блоке или списке параметров функции, если нет никаких специальных указаний:
- Ответ: **auto**.

# Вопрос 3

Дайте ответ на следующие вопросы.

- Вопрос: назовите 4 спецификатора класса памяти:
- Ответ: **auto**, **register**, **extern**, **static**.
- Вопрос: к какому спецификатору класса памяти относится переменная, объявленная в блоке или списке параметров функции, если нет никаких специальных указаний:
- Ответ: **auto**.
- Вопрос: какой спецификатор класса памяти рекомендует компилятору разместить переменную в одном из регистров процессора:

# Вопрос 3

Дайте ответ на следующие вопросы.

- Вопрос: назовите 4 спецификатора класса памяти:
- Ответ: **auto**, **register**, **extern**, **static**.
- Вопрос: к какому спецификатору класса памяти относится переменная, объявленная в блоке или списке параметров функции, если нет никаких специальных указаний:
- Ответ: **auto**.
- Вопрос: какой спецификатор класса памяти рекомендует компилятору разместить переменную в одном из регистров процессора:
- Ответ: **register**.

# Вопрос 3

Дайте ответ на следующие вопросы.

- Вопрос: назовите 4 спецификатора класса памяти:
- Ответ: **auto**, **register**, **extern**, **static**.
- Вопрос: к какому спецификатору класса памяти относится переменная, объявленная в блоке или списке параметров функции, если нет никаких специальных указаний:
- Ответ: **auto**.
- Вопрос: какой спецификатор класса памяти рекомендует компилятору разместить переменную в одном из регистров процессора:
- Ответ: **register**.
- Вопрос: если переменная объявлена вне любого блока или функции, она является:

# Вопрос 3

Дайте ответ на следующие вопросы.

- Вопрос: назовите 4 спецификатора класса памяти:
- Ответ: **auto**, **register**, **extern**, **static**.
- Вопрос: к какому спецификатору класса памяти относится переменная, объявленная в блоке или списке параметров функции, если нет никаких специальных указаний:
- Ответ: **auto**.
- Вопрос: какой спецификатор класса памяти рекомендует компилятору разместить переменную в одном из регистров процессора:
- Ответ: **register**.
- Вопрос: если переменная объявлена вне любого блока или функции, она является:
- Ответ: внешней, глобальной.

# Вопрос 4

Дайте ответ на следующие вопросы.

- Вопрос: с каким спецификатором класса памяти должна быть объявлена локальная переменная, чтобы она сохраняла свое значение между вызовами функции:

# Вопрос 4

Дайте ответ на следующие вопросы.

- Вопрос: с каким спецификатором класса памяти должна быть объявлена локальная переменная, чтобы она сохраняла свое значение между вызовами функции:
- Ответ: **static**.

# Вопрос 4

Дайте ответ на следующие вопросы.

- Вопрос: с каким спецификатором класса памяти должна быть объявлена локальная переменная, чтобы она сохраняла свое значение между вызовами функции:
- Ответ: **static**.
- Вопрос: назовите четыре возможных области действия идентификатора:



# Вопрос 4

Дайте ответ на следующие вопросы.

- Вопрос: с каким спецификатором класса памяти должна быть объявлена локальная переменная, чтобы она сохраняла свое значение между вызовами функции:
- Ответ: **static**.
- Вопрос: назовите четыре возможных области действия идентификатора:
- Ответ: 1. область действия функции, 2. область действия файла, 3. область действия блока, 4. область действия прототипа.

# Вопрос 4

Дайте ответ на следующие вопросы.

- Вопрос: с каким спецификатором класса памяти должна быть объявлена локальная переменная, чтобы она сохраняла свое значение между вызовами функции:
- Ответ: **static**.
- Вопрос: назовите четыре возможных области действия идентификатора:
- Ответ: 1. область действия функции, 2. область действия файла, 3. область действия блока, 4. область действия прототипа.
- Вопрос: как называется функция, которая вызывает саму себя непосредственно или косвенно:

# Вопрос 4

Дайте ответ на следующие вопросы.

- Вопрос: с каким спецификатором класса памяти должна быть объявлена локальная переменная, чтобы она сохраняла свое значение между вызовами функции:
- Ответ: **static**.
- Вопрос: назовите четыре возможных области действия идентификатора:
- Ответ: 1. область действия функции, 2. область действия файла, 3. область действия блока, 4. область действия прототипа.
- Вопрос: как называется функция, которая вызывает саму себя непосредственно или косвенно:
- Ответ: рекурсивная функция.

# Вопрос 5

Для программы, приведенной ниже, установите область действия каждого из перечисленных элементов:

```
int cube(int y);  
main(){  
    int x;  
    for (x=1; x<=10; x++)  
        printf("%d\n", cube(x));  
}  
int cube(int y) { return y = y*y*y;}
```

- переменная x в main:

# Вопрос 5

Для программы, приведенной ниже, установите область действия каждого из перечисленных элементов:

```
int cube(int y);  
main(){  
    int x;  
    for (x=1; x<=10; x++)  
        printf("%d\n", cube(x));  
}  
int cube(int y) { return y = y*y*y;}
```

- переменная x в main:
- Ответ: область действия блока.

# Вопрос 5

Для программы, приведенной ниже, установите область действия каждого из перечисленных элементов:

```
int cube(int y);  
main(){  
    int x;  
    for (x=1; x<=10; x++)  
        printf("%d\n", cube(x));  
}  
int cube(int y) { return y = y*y*y;}
```

- переменная x в main:
- Ответ: область действия блока.
- переменная y в cube:

# Вопрос 5

Для программы, приведенной ниже, установите область действия каждого из перечисленных элементов:

```
int cube(int y);  
main(){  
    int x;  
    for (x=1; x<=10; x++)  
        printf("%d\n", cube(x));  
}  
int cube(int y) { return y = y*y*y;}
```

- переменная x в main:
- Ответ: область действия блока.
- переменная y в cube:
- Ответ: область действия блока.

# Вопрос 5 (продолжение)

- функция `main`:



# Вопрос 5 (продолжение)

- функция main:
- Ответ: область действия файла.

# Вопрос 5 (продолжение)

- функция `main`:
- Ответ: область действия файла.
- функция `cube`:

# Вопрос 5 (продолжение)

- функция main:
- Ответ: область действия файла.
- функция cube:
- Ответ: область действия файла.

# Вопрос 5 (продолжение)

- функция `main`:
- Ответ: область действия файла.
- функция `cube`:
- Ответ: область действия файла.
- прототип функции `cube`:

# Вопрос 5 (продолжение)

- функция main:
- Ответ: область действия файла.
- функция cube:
- Ответ: область действия файла.
- прототип функции cube:
- Ответ: область действия файла.

# Вопрос 5 (продолжение)

- функция `main`:
- Ответ: область действия файла.
- функция `cube`:
- Ответ: область действия файла.
- прототип функции `cube`:
- Ответ: область действия файла.
- переменная `y` в прототипе функции `cube`:

# Вопрос 5 (продолжение)

- функция `main`:
- Ответ: область действия файла.
- функция `cube`:
- Ответ: область действия файла.
- прототип функции `cube`:
- Ответ: область действия файла.
- переменная `y` в прототипе функции `cube`:
- Ответ: область действия прототипа.

# Вопрос 6

Напишите следующие объявления:

- переменная `count` целого типа, которая должна обрабатываться в регистре и иметь начальное значение 0



# Вопрос 6

Напишите следующие объявления:

- переменная `count` целого типа, которая должна обрабатываться в регистре и иметь начальное значение 0
- Ответ: `register int count = 0;`

# Вопрос 6

Напишите следующие объявления:

- переменная `count` целого типа, которая должна обрабатываться в регистре и иметь начальное значение 0
- Ответ: `register int count = 0;`
- вещественная переменная `val`, которая сохраняет свое значение между вызовами функции:

# Вопрос 6

Напишите следующие объявления:

- переменная `count` целого типа, которая должна обрабатываться в регистре и иметь начальное значение 0
- Ответ: `register int count = 0;`
- вещественная переменная `val`, которая сохраняет свое значение между вызовами функции:
- Ответ: `static float val;`

# Вопрос 6

Напишите следующие объявления:

- переменная `count` целого типа, которая должна обрабатываться в регистре и иметь начальное значение 0
- Ответ: `register int count = 0;`
- вещественная переменная `val`, которая сохраняет свое значение между вызовами функции:
- Ответ: `static float val;`
- переменная `number` целого типа, чья область действия должна ограничиваться оставшейся частью файла, в которой определена переменная:

# Вопрос 6

Напишите следующие объявления:

- переменная `count` целого типа, которая должна обрабатываться в регистре и иметь начальное значение 0
- Ответ: `register int count = 0;`
- вещественная переменная `val`, которая сохраняет свое значение между вызовами функции:
- Ответ: `static float val;`
- переменная `number` целого типа, чья область действия должна ограничиваться оставшейся частью файла, в которой определена переменная:
- Ответ: `static int number` (объявление должно находиться вне любого определения функции).

# Вопрос 7

Найдите ошибку в фрагменте кода и объясните, как ее исправить:

```
int g(viod)
{
    printf("Hello, it is function g \n");
    int h(void)
    {
        printf("Hello, again. It is function h \n");
    }
}
```

# Вопрос 8

Найдите ошибку в фрагменте кода и объясните, как ее исправить:

```
int sum(int x, int y)
{
    int result;
    result = x+y;
}
```

# Вопрос 9

Найдите ошибку в фрагменте кода и объясните, как ее исправить:

```
int sum(int n)
{
    if (n == 0)
        return 0;
    else
        n+sum(n-1);
}
```



# Вопрос 10

Найдите ошибку в фрагменте кода и объясните, как ее исправить:

```
void f(float a);  
{  
    float a;  
    printf(" %d", a);  
}
```

# Вопрос 11

Найдите ошибку в фрагменте кода и объясните, как ее исправить:

```
void product(void)
{
    int a,b,c, result;
    printf("Enter three numbers");
    scanf("%d%d%d", &a, &b, &c);
    result = a*b*c;
    printf("Result is %d", result);
    return result;
}
```