

Надо сделать: Получить оценку

Лабораторная работа №6 Функции

Цель работы: Изучение методов использования функций языка

Основные сведения

Часто в программе требуется повторить определенную последовательность операторов в разных частях программы. Для того, чтобы описывать эту последовательность один раз, а применять многократно, в языках программирования применяются подпрограммы. Подпрограмма - автономная часть программы, выполняющая определенный алгоритм и допускающая обращение к ней из различных частей общей программы.

В языке Си существует один вид подпрограмм - функции. Каждая программа в своем составе должна иметь главную функцию `main()`, служащую точкой входа в программу. Кроме функции `main()`, в программу может входить произвольное число функций, выполнение которых инициализируется либо прямо, либо вызовами из функции `main()`. Каждая функция по отношению к другой является внешней. Для того, чтобы функция была доступной, необходимо, чтобы до ее вызова о ней было известно компилятору. Форма записи функции:

<тип> <имя функции>(<формальные параметры>){<тело функции>}

Если тип возвращаемого функцией значения не указан, то подразумевается `int`. Если с именем функции не связан результат, то нужно указать тип функции `void`. Параметры, записываемые в обращении к функции, называются фактическими; параметры, указанные в описании функции - формальными. Фактические параметры должны соответствовать формальным по количеству, порядку следования и типу. Объекты, объявленные вне функции, действуют в любой функции и называются глобальными. Объекты, объявленные в функции, действуют только в ней и называются локальными. В теле функции обычно присутствует оператор `return <выражение>`, определяющий возвращаемое функцией значение.

Все параметры функции, кроме массивов, передаются по значению, т.е. внутри функции создаются локальные копии параметров. Если необходимо передать саму переменную, а не её копию, то в функцию передаётся адрес этой переменной. Таким образом, через параметры можно передавать результат выполнения функции. То есть, параметры, с помощью которых результаты должны передаваться из функции в точку вызова, описываются как указатели. Вызов функции может быть оформлен в виде оператора, если с именем функции не связано возвращаемое значение, или в виде выражения, если возвращаемое значение связано с именем функции.

Прототип функции может указываться до вызова функции вместо описания функции для того, чтобы компилятор мог выполнить проверку соответствия типов аргументов и параметров. Прототип функции по форме такой же, как и заголовок функции. В конце него ставится «;».

Функции можно подключать с помощью директивы `#include <имя файла>`. Такие файлы с функциями удобно использовать в диалоговых программах с пользовательским меню, позволяющих выбрать один из режимов.

Пример 1: Функция с параметрами-значениями. Результат связан с именем функции. В программе объявляется прототип функции, а сама функция описывается ниже.



```

//lab8_1
#include <stdio.h>
#include <conio.h>
int max(int,int); //Прототип функции
void main()
{ int x,y,z;
    printf(" input x,y ");
    scanf("%d%d",&x,&y);
    z=max(x,y); //Вызов функции с фактическими параметрами
    printf("x=%d y=%d max=%d",x,y,z);
    getch();
}
int max(int a ,int b) //Заголовок функции с формальными параметрами
{ int c;
    if (a>b) c=a;
    else c=b;
    return c;
}

```

Пример 2: Функция с параметрами-указателями. Здесь передаются адреса фактических параметров, по которым и получаем результат. Функция меняет местами переменные x,y.

```

//lab8_2
#include <stdio.h>
#include <conio.h>
main()
{ float x,y;
    void swap(float *, float *); // Прототип функции с параметрами - указателями
    printf("\n введите x,y ");
    scanf("%f%f",&x,&y);
    swap(&x,&y); // Передаём адреса переменных
    printf("\n x=%4.2f y=%4.2f ",x,y);
    getch();
}
void swap(float * a, float * b)
{float c;
    c=*a; // *a - содержимое по адресу a
    *a=*b;
    *b=c;
}

```

Пример 3: Подключение файлов с функциями и создание меню.

```

//lab8_3
#include <stdio.h>
#include <conio.h>
#include "lab3.c"
#include "lab5.c"
#include "lab6.c"
main()
{ int nom;
    while(1)
    { clrscr();
        printf("\n 1. Сумма ряда \n");
        printf(" 2. Матрица \n");
        printf(" 3. Строки \n");
        printf(" 4. Выход \n");
        scanf("%d",&nom);
        switch(nom)
        {
            case 1:lab3();break;
            case 2:lab5();break;
            case 3:lab6();break;
            case 4:return 0;
            default:printf("Неверный режим");
        }
    }
    getch();
}

```

! Внимание! Следите за тем, чтобы константы, объявленные директивой #define, не переобъявлялись в функциях.

Пример 4: Передача в функцию массива с использованием указателя. Результат – элементы массива возводятся в квадрат. Функция описывается до вызова, поэтому прототип не объявляется.

```

//lab8_4
#include <stdio.h>
#include <conio.h>
void quart(int n, float * x) // Можно void quart(int n, float x[])
{ int i;
  for (i=0;i<n;i++)
    x[i]=x[i]*x[i];
}
main()
{ float z[]={1,2,3,4};int j;
 clrscr();
 for (j=0;j<4;j++)
 printf("%6.2f",z[j]);
 quart(4,z);
 for (j=0;j<4;j++)
 printf("\n %6.2f",z[j]);
 getch();
}

```

Выполнение работы

1. Проанализировать приведенные выше программы.
2. Оформить свои задания по лабораторным работам 5, 6, 7 в виде функций. Организовать меню с вызовом этих функций.
3. Используя функции, написать программу по своему варианту.

Варианты заданий

1. Написать функцию, выводящую в порядке возрастания элементы одномерного массива. В главной программе вызвать функцию для двух разных массивов.
2. Написать функцию вычисления произведения прямоугольной матрицы A размера k x m на прямоугольную матрицу B размера m x n. В главной программе обратиться к этой функции.
3. Написать функцию вычисления суммы ряда $s=s(1)+\dots+s(n)$, где $s(n)=(-1)^n \frac{x^{(2n-1)}}{(2n+1)}$ с точностью до $\text{eps}=0.001$. В качестве параметров выбрать x и eps.
4. Написать функцию, которая вычисляет для заданной квадратной матрицы A её симметричную часть $S(ij)=(A(ij)+A(ji))/2$ и кососимметричную часть $K(ij)=(A(ij)-A(ji))/2$.
5. Написать функцию “шапочка” $f(x)$, зависящую от параметров a и b: если $|x| > a$ то $f(x)=0$ иначе $f(x)=b*\exp(-a^2/(a^2-|x|^2))$. В качестве параметров передать a,b,x.
6. Написать функцию поиска максимального и минимального элементов одномерного массива. В основной программе вызвать эту функцию для двух разных массивов.
7. Написать функцию, которая сортирует одномерный массив в порядке убывания методом пузырька. В основной программе вызвать эту функцию для двух разных массивов.
8. Написать функцию, которая по двум заданным одномерным массивам (A размера m и B размера n) вычисляет максимальное значение двумерного массива $c(ij)=a(i)*b(j)$.
9. Написать функцию определителя квадратной матрицы A размера 3x3: $\det A = a(1,1)a(2,2)a(3,3) + a(3,1)a(1,2)a(2,3) + a(2,1)a(3,2)a(1,3) - a(3,1)a(2,2)a(1,3) - a(1,1)a(3,2)a(2,3) - a(2,1)a(1,2)a(3,3)$.
10. Написать функцию вычисления суммы ряда $y=\sin x - (\sin 2x)/2 + \dots + (-1)^{n+1} \sin(nx)/n$ с точностью до $\text{eps}=0.001$. В качестве параметров передать x (в радианах) и eps.
11. Написать функцию вычисления ряда $y=x+x^3/3!+\dots+x^{2n+1}/(2n+1)!$ с точностью до $\text{eps}=0.0001$. В качестве параметров передать x и eps.
12. Написать функцию обработки матриц A и B одинакового размера m x n. Получить матрицу C = max(a(i,j),b(i,j)), и матрицу D = min(a(i,j),b(i,j)). Матрицы C и D вывести в главной программе.

Контрольные вопросы

1. Описание функции. Для чего объявляется прототип?
2. Что такое формальные и фактические параметры? Локальные и глобальные?
3. Как можно передавать массив в функцию?
4. Способы вызова функций.

[Добавить ответ на задание](#)

Состояние ответа

| | |
|------------------------------------|---------------------------------------|
| Состояние ответа на задание | Ответы на задание еще не представлены |
| Состояние оценивания | Не оценено |

Информация

Официальный сайт ФГБОУ ВО
Белгородский ГАУ

Личный кабинет преподавателя
и студента

Расписание

Отдел электронных
образовательных ресурсов и
сетевого обучения

Структура университета

Контакты

308503, Белгородская обл.,
Белгородский р-н, п. Майский, ул.
Вавилова, 1, отдел электронных
образовательных ресурсов и
сетевого обучения, №321 (с 8.00 до
17.00, перерыв 12.00-13.00)

 Телефон : +7 (4722) 39-22-51 (по
вопросам ЭИОС). По вопросам
справок: +7 (4722) 38-05-17 (МФЦ
БелГАУ)

 Эл.почта : help@belgau.ru

© 2025 Белгородский государственный аграрный университет имени В.Я. Горина

