

---

## Лабораторная работа.2. Управляющие инструкции, массивы

### Цель выполнения лабораторной работы

Освоение управляющих инструкций, и работы с массивами данных в языке Си.

### Порядок выполнения работы

Обучаемые разрабатывают программы в соответствии с представленным описанием по выданным им индивидуальным заданиям.

### Управляющие инструкции *if* и *switch*

Рассмотрим пример использования управляющих инструкций.

#### Задание 1:

Написать программу, которая позволяет посчитать цену жалюзи. Исходные данные: размер (ширина и высота, выраженные в сантиметрах) и тип материала (пластик, текстиль, алюминий). Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным).

```
Jalousie
Width (см) -    ->    75
Height (см) -  ->    150
Material:
1 - Plastic
2 - Textiles
3 - Aluminum
Your choice  -> 3

Price for kv. m: 350.00 rub.
Area: 1.13 kv. m.
To payment: 393.75 rub.
```

#### Код программы:

```
// Жалюзи - вычисляет цену жалюзи
// в зависимости от размера и материала
#include "stdio.h"
#include "conio.h"
void main()
{
    float w,h; // ширина, высота
    int m;     // материал:
    // 1 - пластик; 2 - текстиль; 3 - алюминий;
    float cena; // цена за 1 кв. м.
    float s;    // площадь
    float sum;  // сумма

    printf("\nJalousie\n");
    printf("\nWidth  (см.) -> ");
    scanf("%f",&w);
    printf("Height (см.) -> ");
    scanf("%f",&h);

    printf("Metherial\n");
```

```

printf("1 - Plastic\n");
printf("2 - Textiles\n");
printf("3 - Aluminum\n");
printf("\n Your choice  ->");

scanf("%i",&m);
// выбор материала
switch ( m )
{
    case 1: cena = 200; break;
    case 2: cena = 250; break;
    case 3: cena = 350; break;

    default: cena = 0; break;
}
// если выбор правильный то вычисляем цену
if ( cena != 0 )
{
    s = (w * h) / 10000;
    sum = s * cena;
    printf("\nPrice for kv. m.: %3.2f rub", cena);
    printf("\nArea: %3.2f кв.м.", s);
    printf("\nTo payment: %3.2f rub", sum);
}
else // если выбор не правильный, выдаем сообщение
    printf("\nWrong a code of the material is specified!");

printf("\n For termination press <Enter>\n");
getch();
}

```

**Скомпилируйте программу вводом команды:**

```
>cl <имя_файла.c> /Fe<имя_исполняемого_файла.exe>
```

**Для нашего примера получится следующая команда:**

```
>cl Jolusie.c /FeJolusie.exe
```

Для вызова программы на исполнение введите имя исполняемого файла.

### Общие замечания

Приступая к решению задач этого раздела, следует вспомнить:

Инструкция `if` используется для выбора одного из двух направлений дальнейшего хода программы;

Выбор действия (последовательности инструкций) осуществляется в зависимости от значения условия — заключенного в скобки выражения, записанного после `if`.

Инструкция, записанная после `else`, выполняется в том случае, если значение выражения условие равно нулю, во всех остальных случаях выполняется инструкция, следующая за условием.

Если при выполнении (или невыполнении) условия требуется выполнить несколько инструкций программы, то эти инструкции следует объединить в группу — заключить в фигурные скобки.

При помощи вложенных одна в другую нескольких инструкций `if` можно реализовать множественный выбор.

Про `switch` следует помнить следующее:

Инструкция `switch` используется для выбора одного из нескольких возможных вариантов дальнейшего хода программы.

---

Выбор действия осуществляется в зависимости от равенства значения переменной-селектора к константе, указанной после слова case.

Если значение переменной-селектора не равно ни одной из констант, записанных после case, то выполняются инструкции, записанные после слова default.

В качестве переменной-селектора можно использовать переменную целого (int) или символьного (char) типа.

### Индивидуальное задание №1

Создание программы с управляющей инструкции if.

Варианты заданий

1. Написать программу вычисления сопротивления электрической цепи, состоящей из двух сопротивлений, которые могут быть соединены последовательно или параллельно. Ниже приведен рекомендуемый вид экрана (данные, введенные пользователем, выделены полужирным).

Вычисление сопротивления электрической цепи

Введите исходные данные:

Величина первого сопротивления (Ом) -> **15**

Величина второго сопротивления (Ом) -> **27.3**.

Тип соединения (1-последовательное, ' 2 - параллельное) ' "-> **2**

Сопротивление цепи: 9.68 Ом

2. Написать программу вычисления дохода по вкладу. Исходные данные: сумма и срок вклада. Процентная ставка зависит от суммы. Если сумма меньше 5000 руб., то процентная ставка 10%, если больше, то 13%. Ниже приведен рекомендуемый вид экрана (данные, введенные пользователем, выделены полужирным).

Доход

Сумма, руб. -> **10000**

Срок вклада, мес. -> **12**

Сумма: 10000.00 руб.

Срок вклада: 12 мес.

Процент годовой: 13

Доход: 1300.00 руб.

Сумма в конце срока вклада: 11300.00 руб.

3. Написать программу вычисления дохода по вкладу. Исходные данные: сумма и срок вклада. Процентная ставка зависит от суммы. Если сумма меньше 5000 руб., то процентная ставка 9%, если больше 5000 руб., но меньше 10 000 руб., то 11%, а если больше 10 000, то 13%. Ниже приведен рекомендуемый вид экрана (данные, введенные пользователем, выделены полужирным).

Доход

Сумма, руб. -> **15000**

Срок вклада, мес. -> **6**

Сумма: 15000.00 руб.

Срок вклада: 6 мес.

Процент годовой: 13.00

Доход: 975.00 руб.

Сумма в конце срока вклада: 15975.00 руб.

4. Написать программу вычисления стоимости печати фотографий. Формат фотографий 9x12 или 10x15. Если количество фотографий больше 10, то заказчику предоставляется скидка 5%. Ниже приведен рекомендуемый вид экрана (данные, введенные пользователем, выделены полужирным).

Фотографии-  
Формат (1 - 9x12; 2 - 10x1.5) -> 2  
Количество, шт. -> 15  
  
Цена: 3.20 руб.  
Количество: 15 шт.  
Сумма: 48.00 руб.  
Скидка: 2.40 руб.  
К оплате: 45.60 руб.

5. Написать программу, которая вычисляет величину тока, потребляемого электроприбором ( $I = P/U$ , где:  $I$ — ток, (А);  $P$ — мощность, (Вт);  $U$ — напряжение, (В)). Программа должна проверять правильность введенных пользователем данных и, если они неверные (делитель равен нулю), выводить сообщение об ошибке. Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным).

Ток в электрической цепи  
Мощность, Вт -> 60  
Напряжение, В -> 0

Ошибка! Напряжение не должно быть равно нулю.

6. Написать программу вычисления площади кольца. Программа должна проверять правильность исходных данных. Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным).

Вычисление площади кольца  
Введите исходные данные:  
радиус кольца (см) -> 3.5  
радиус отверстия (см) -> 7

Ошибка! Радиус отверстия больше радиуса кольца

7. Написать программу, которая переводит время из минут и секунд в секунды. Программа должна проверять правильность введенных пользователем данных и в случае, если данные неверные, выводить соответствующее сообщение; Рекомендуемый вид экрана программы приведен ниже. Ошибочные Данные, введенные пользователем, выделены полужирным.

Введите время (минут .секунд) -> 2.90

Ошибка! Число секунд не может быть больше 60  
Для завершения нажмите <Enter>

8. Написать программу, которая проверяет, является ли год високосным. Ниже приведен рекомендуемый вид экрана программы. Данные, введенные пользователем, выделены полужирным.

Введите год, например 2000, и нажмите <Enter>  
->2001  
2001 год - не високосный

Для завершения нажмите <Enter>

9. Написать программу решения квадратного уравнения. Программа должна проверять правильность исходных данных и в случае, если коэффициент при второй степени неизвестного равен нулю, выводить соответствующее сообщение.

Вычисление дискриминанта:

$$d = b*b - 4*a*c;$$

Если дискриминант меньше нуля то уравнение не имеет решений.

Вычисление корней уравнения:

$$x(1/2) = (-b (+/-) \sqrt{d})/(2*a) ;$$

Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным).

```
Решение квадратного уравнения
Введите в одной строке коэффициенты и нажмите <Enter>
-> 12 27 -10
Корни уравнения:
x1= -25.551
x2= -28.449
```

10. Написать программу вычисления стоимости покупки с учетом скидки. Скидка в 10% предоставляется, если сумма покупки больше 1000 руб. Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным).

```
Вычисление стоимости покупки с учетом скидки
Сумма покупки -> 1200
Вам предоставляется скидка 10%
Сумма покупки с учетом скидки: 1080.00 руб.
```

11. Написать программу вычисления стоимости покупки с учетом скидки. Скидка в 3% предоставляется, если сумма покупки больше 500 руб, в 5% — если сумма больше 1000 руб. Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным).

```
Вычисление стоимости покупки с учетом скидки
Введите сумму покупки и нажмите <Enter> -> 640
Вам предоставляется скидка 3%
Сумма с учетом скидки: 620.80 руб.
```

12. Написать программу проверки знания даты основания Санкт-Петербурга. В случае неправильного ответа пользователя, программа должна выводить правильную дату. Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным).

```
В каком году был основан Санкт-Петербург?
Введите число и нажмите <Enter>
-> 1705
Вы ошиблись, Санкт-Петербург был основан в 1703 году
```

13. Написать программу проверки знания даты начала Второй мировой войны. В случае неправильного ответа пользователя, программа должна выводить правильную дату. Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным).

```
В каком году началась Вторая мировая война?
```

Введите число и нажмите <Enter;>  
-> **1939**  
Правильно!

14. Написать программу проверки знания истории архитектуры. Программа должна вывести вопрос и три варианта ответа. Пользователь должен выбрать правильный ответ и ввести его номер. Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным).

Архитектор Исаакиевского собора:  
1. Доменико Трезини  
2. Огюст Монферран  
3. Карл Росси  
Введите номер правильного ответа и нажмите <Enter>  
-> **3**  
Вы ошиблись.  
Архитектор Исаакиевского собора - Огюст Монферран

15. Написать программу проверки знания истории архитектуры. Программа должна вывести вопрос и три варианта ответа. Пользователь должен выбрать правильный ответ и ввести его номер. Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным).

Невский проспект получил свое название:  
1. по имени реки, на берегах которой расположен Санкт-Петербург  
2. по имени близко расположенного монастыря Александро-Невской лавры  
3. в память о знаменитом полководце Александре Невском  
Введите номер правильного ответа и нажмите <Enter>  
-> **1**  
Вы ошиблись. Правильный ответ: 2

16. Написать программу, которая сравнивает два введенных с клавиатуры числа. Программа должна указать, какое число больше, или, если числа равны, вывести соответствующее сообщение. Ниже приведен рекомендуемый вид экрана программы.

Введите в одной строке два целых числа и нажмите <Enter>  
-> 34 67  
34 меньше 67

17. Написать программу, которая выводит пример на умножение двух однозначных чисел, запрашивает ответ пользователя, проверяет его и выводит сообщение "Правильно!" или "Вы ошиблись" и правильный результат. Ниже приведен рекомендуемый вид экрана программы (данные, введенные Пользователем, выделены полужирным).

Сколько будет 6х7 ?  
Введите ответ и нажмите <Enter>  
-> **46**  
Вы ошиблись. 6х7=42

18. Написать программу, которая проверяет, является ли введенное пользователем целое число четным. Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным).

Введите целое число и нажмите <Enter>  
-> 23  
Число 23 - нечетное.

19. Написать программу, которая проверяет, делится ли на три введенное с клавиатуры целое число. Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным).

Введите целое число и нажмите <Enter> -> **451**  
451 на три не делится

20. Написать программу вычисления стоимости разговора по телефону с учетом 20%-ной скидки, предоставляемой по субботам и воскресеньям. Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным).

Вычисление стоимости разговора по телефону  
Введите исходные данные:  
Длительность разговора (целое количество минут) -> 3  
День недели (1 - понедельник, ... 7 - воскресенье) -> 6  
Предоставляется скидка 20%. Стоимость разговора: 5.52 руб.

## Индивидуальное задание №2

Для выполнения задания 2 добавьте условие в задание №1 для того чтобы актуально стало использование управляющей инструкции switch. Или возьмите одно из ниже представленных заданий.

Пример расширения задания 2:

### Исходное задание

Написать программу вычисления стоимости разговора по телефону с учетом 20%-ной скидки, предоставляемой по субботам и воскресеньям.

### Расширенное задание:

Написать программу вычисления стоимости разговора по телефону с учетом системы скидок. Скидки, предоставляются следующим образом:

по субботам. – 20%  
по воскресеньям - 30%  
по средам – 10%  
по четвергам – 5%.

Создание программы с управляющей инструкции switch:

1. Написать программу, которая позволяет посчитать стоимость печати фотографий. Исходные данные: размер фотографий (9x12, 10x15 или 18x24) и их количество. Если заказанных фотографий больше 10, заказчику должна предоставляться скидка 10%. Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным).

Фото Размер:  
1 - 9x12  
2 - 10x15  
3 - 18x24  
Ваш выбор -> **1**  
Количество -> **12**

Цена: 3.50 руб.  
Количество: 12 шт.  
Сумма: 42.00' руб.  
Скидка: 4.20 руб.  
К оплате: 37.80 руб.

2. Написать программу, которая запрашивает у пользователя номер дня недели и затем выводит его название. Если введены неправильные данные, программа должна вывести сообщение об ошибке.
3. Написать программу, которая вычисляет доход по вкладу. Процентная ставка зависит от срока вклада:

Срок	3	6	12	18	24	36
	мес.	мес.	мес.	мес.	мес.	мес.
Процент	9,0%	11,5	13,5	15,0	18,0	24,0
		%	%	%	%	%

4. Написать программу, которая вычисляет стоимость междугородного телефонного разговора (цена одной минуты зависит от расстояния до города, в котором находится абонент). Исходные данные для программы: код города и длительность разговора. Ниже приведены коды некоторых городов, и рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным).

Город	Код	Цена минуты (руб.)
Владивосток	423	2,2
Москва	495	1,0
Мурманск	815	1,2
Самара	846	1,4

Вычисление стоимости разговора по телефону  
Введите исходные данные:  
Код города -> 423  
Длительность (минут) -> 3  
Город: Владивосток  
Цена минуты: 2.20 руб.  
Стоимость разговора: 6.60 руб.

## Управляющие конструкции циклов

### Цикл for

Цикл for, наиболее универсальный из всех циклов языка C, выглядит так:

```
for ([инициализация]; [условие]; [модификация])  
{  
    операторы  
}
```

Прежде всего, выполняется инициализация цикла; секция инициализации может содержать любое выражение. Инициализация производится только один раз перед началом работы цикла.

Оценивается выражение условия. Если оно истинно, выполняется оператор тела цикла; если условие ложно, происходит выход из цикла и управление передается следующему оператору.

После исполнения тела цикла производится модификация, после чего управление возвращается заголовку цикла и все повторяется снова. Секция



модификации может содержать любое выражение; обычно в ней изменяют значения управляющих переменных цикла.

Простейшей и самой популярной конструкцией на основе цикла for является цикл с управляющей переменной-счетчиком:

```
int i;
for (i = 0; i < REPEAT; i++)
    doSomething(i);
```

Приступая к решению задач индивидуального задания, следует вспомнить, что

Инструкция for используется для организации циклов с фиксированным числом повторений.

Количество повторений цикла определяется начальным значением переменной-счетчика и условием завершения цикла.

Переменная-счетчик должна быть целого типа и может быть объявлена непосредственно в инструкции цикла.

### Задание 2:

Написать программу, которая выводит таблицу значений функции  $y = -2,4x^2 + 5x - 3$  в диапазоне от -2 до 2, с шагом 0,5. Ниже приведен рекомендуемый вид экрана программы.

x		y
-2		-22.60
-1.5		-15.90
-1		-10.40
-0.5		-6.10
0		-3.00
0.5		-1.10
1		-0.40
1.5		-0.90
2		-2.60

Запишите и скомпилируйте код реализации программы с применением цикла for. Проверьте правильность исполнения.

```
// Таблица функции
#include <stdio.h>
#include <conio.h>

#define LB -2.0 // нижняя граница диапазона изменения аргумента
#define HB 2.0 // верхняя граница диапазона изменения аргумента
#define DX 0.5 // приращение аргумента

void main()
{
    float x,y; // аргумент и значение функции
    int n; // кол-во точек
    int i; // счетчик циклов

    n = (HB - LB)/DX +1;
    x = LB;
    printf("-----\n");
    printf("    x    |    y\n");
    printf("-----\n");
    for (i = 1; i<=n; i++)
```

```

    {
        y = -2.4*x*x+5*x-3;
        printf("%6.2f | %6.2f\n" , x , y);
        x += DX;
    }
    printf("-----\n");

    printf("\nPress <Enter> for finish");
    getch();
}

```

### Цикл while и do... while

Реализация представленного задания возможна также с применением цикла do ... while или while.

Приступая к решению задач этого раздела, следует вспомнить, что цикл do... while:

Цикл do while — это цикл с постусловием, т. е. инструкции тела цикла (между do и while) будут выполнены хотя бы один раз.

После слова while записывается условие повторного выполнения инструкций цикла.

Число повторений инструкций цикла do while определяется ходом выполнения программы.

Для завершения цикла do while в теле цикла обязательно должны быть инструкции, выполнение которых влияет на условие завершения цикла.

Цикл do while, как правило, используется для организации приближенных вычислений, в задачах поиска и обработки данных, вводимых с клавиатуры или из файла.

Цикл while:

while — это цикл с предусловием, т.е. возможна ситуация, при которой инструкции тела цикла не будут выполнены ни разу.

Инструкции цикла while выполняются до тех пор, пока условие истинно (значение выражения Условие не равно нулю).

Для того чтобы цикл while завершился, в его теле должны быть инструкции, влияющие на значения переменных, входящих в условие выполнения цикла.

Цикл while обычно используется для организации приближенных вычислений, в задачах поиска и обработки данных вводимых с клавиатуры или из файла.

### Задание 3:

Введите, скомпилируйте и проверьте реализацию задачи примера из задания №2 но с использование цикла while:

```

// Таблица функции
#include <stdio.h>
#include <conio.h>

void main()
{
    float x,y; // аргумент и значение функции

```

```

float x1,x2,dx; // диапазон и шаг изменения аргумента

printf("x1 -> ");
scanf("%f",&x1);
printf("x2 -> ");
scanf("%f",&x2);
printf("dx -> ");
scanf("%f",&dx);

x = x1;
printf("-----\n");
printf("    x    |    y\n");
printf("-----\n");
do
{
    y = -2.4*x*x+5*x-3;
    printf("%6.2f    | %6.2f\n" ,x ,y);
    x += dx;
}
while ( x <= x2 );

printf("-----\n");
printf("\nPress <Enter> for finish\n");
getch();
}

```

### Индивидуальное задание №3

1. Написать программу, которая выводит на экран ваше имя 10 раз.
2. Написать программу, которая выводит таблицу квадратов первых десяти целых положительных чисел. Ниже приведен рекомендуемый вид экрана программы.

Таблица квадратов	
Число	Квадрат
1 ;	1
2	4
3	9
4	16
5	25
6	36
7	49
8	64
9	81
10	100

3. Написать программу, которая выводит таблицу квадратов первых пяти целых положительных нечетных чисел. Ниже приведен рекомендуемый вид экрана программы.

Таблица квадратов нечетных чисел.	
Число	Квадрат
1	1
3	9
5	25
7	49

9	81
---	----

4. Написать программу, которая выводит таблицу скорости (через каждые 0,5 с) свободно падающего тела ( $v = g \cdot t$ , где  $g = 9,8$  м/с — ускорение свободного падения). Рекомендуемый вид экрана приведен ниже.

Время, с	Скорость, м/с
0.0	0.00
0.5	4.90
1.0	9.80
1.5	14.70
2.0	19.60
2.5	24.50
3.0	29.40

5. Написать программу, которая выводит таблицу ежемесячных платежей по кредиту. Исходные данные для расчета: сумма кредита, срок и процентная ставка. Предполагается, что кредит возвращается (выплачивается) ежемесячно равными долями. Проценты начисляются ежемесячно на величину долга. Рекомендуемый вид экрана приведен иже.

Сумма (руб.) -> 150000				
Срок (мес.) -> 12				
Процентная ставка (годовых) -> 14				
До	Процент	Платеж		
лг				
150000	17	.	1425	.
.00	50	00	0	00
137500	16	.	1410	.
.00	04	17	4	00
125000	14	.	1395	.
.00	58	33	8	33
112500	13	.	1381	.
.00	12	50	2	50
100000	11	.	1366	.
.00	66	67	6	67
87500.	10	.	1352	.
00	20	83	0	83
75090.	87	.	1337	.
00	5	00	5	00
62500.	72	.	1322	.
00	9	17	9	17
50000.	58	.	1308	.
00	3	33	3	33
37500.	43	.	1293	.
0 00	7	50	7	50
25000.	29	.	1279	.
1 00	1	67	1	67
12500.	14	.	1264	.
2 00	5	83	5	83
Всего процентов: 11375.00				

6. Написать программу, которая выводит на экран таблицу соответствия температуры в градусах Цельсия и Фаренгейта ( $F^{\circ} = 5/9 \cdot C^{\circ} + 32$ ). Диапазон изменения температуры в градусах Цельсия и

шаг должны вводиться во время работы программы. Рекомендуемый вид экрана приведен ниже.

```
t1 -> 0
t2 -> 10
dt-> 1
```

C	F
0.00	32.00
1.00	33.80
2.00	35.60
3.00	37.40
4.00	39.20
5.00	41.00
6.00	42.80
7.00	44.60
8.00	46.40
9.00	48.20
10.00	50.00

7. Написать программу, которая выводит на экран таблицу перевода длины из дюймов в миллиметры (1 дюйм = 2,54 см). Диапазон длины в дюймах и шаг изменения должны вводиться во время работы программы.
8. Написать программу, которая вычисляет сумму первых  $n$  положительных целых чисел. Количество суммируемых чисел должно вводиться во время работы программы. Ниже приведен рекомендуемый вид экрана (данные, введенные пользователем, выделены полужирным).

```
Сумма положительных чисел
Введите количество суммируемых чисел -> 20
Сумма-первых 20 положительных чисел равна 210
```

9. Написать программу, которая вычисляет сумму первых  $n$  целых положительных четных чисел. Количество суммируемых чисел должно вводиться во время работы программы. Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным).

```
Вычисление суммы четных положительных чисел
Введите количество суммируемых чисел и нажмите <Enter>
-> 12
Сумма первых 12 положительных четных чисел равна 156
```

10. Написать программу, которая вычисляет сумму первых  $n$  членов ряда: 1, 3, 5, 7 ... Количество суммируемых членов ряда задается во время работы программы. Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным).

```
Вычисление частичной суммы ряда: 1, 3, 5, 7, ...
Введите количество суммируемых членов ряда -> 15
Сумма первых 15 членов ряда равна 330
```

11. Написать программу, которая вычисляет сумму первых  $n$  членов ряда:  $1 + 1/2 + 1/3 + 1/4 + \dots$  Количество суммируемых членов ряда задается во время работы программы. Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным).

Вычисление частичной суммы ряда:  $1+1/2+1/3+ \dots$   
Введите кол-во суммируемых членов ряда ->**15**  
Сумма первых 15 членов ряда равна 3.3182

12. Написать программу, которая выводит таблицу степеней двойки, от нулевой до десятой. Ниже приведен рекомендуемый вид экрана программы.

Таблица степеней двойки	
0	1
1	2
2	4
3	8
4	16
5	32
6	64
7	128
8	256
9	512
10	1024

13. Написать программу, которая вводит с клавиатуры пять дробных чисел и вычисляет их среднее арифметическое. Рекомендуемый вид экрана программы приведен ниже. Данные, введенные пользователем, выделены полужирным.

Среднее арифметическое последовательности дробных чисел. После ввода каждого числа нажимайте <Enter>  
-> **5.4**  
-> **7.8**  
-> **3.0**  
-> **1.5**  
-> **2.3**  
Среднее арифметическое введенной последовательности: 4.00  
Для завершения нажмите <Enter>

14. Написать программу, которая вычисляет среднее арифметическое вводимой с клавиатуры последовательности дробных чисел. Количество чисел должно задаваться во время работы программы. Рекомендуемый вид экрана программы приведен ниже.

Вычисление среднего арифметического последовательности дробных чисел.  
Введите количество чисел последовательности -> **3**  
Вводите последовательность. После ввода каждого числа нажимайте <Enter>  
-> **5.4**  
-> **3.0**  
-> **2.3**  
Среднее арифметическое введенной последовательности: 4.00  
Для завершения нажмите <Enter>

15. Написать программу, которая вычисляет среднее арифметическое последовательности дробных чисел, вводимых с клавиатуры. После ввода последнего числа, программа должна вывести минимальное и максимальное числа последовательности. Количество чисел последовательности должно задаваться во время работы программы. Рекомендуемый вид экрана приведен ниже. Данные, введенные пользователем, выделены полужирным.

Обработка последовательности дробных чисел.  
Введите количество чисел последовательности -> **5**  
Вводите последовательность. После ввода каждого числа нажимайте <Enter>

```

-> 5.4
-> 7.8
-> 3.0
-> 1.5
-> 2.3
Количество чисел: 5
Среднее арифметическое: 4.00
Минимальное число: 1.5
Максимальное число: 7.8
Для завершения нажмите <Enter>

```

16. Написать программу, которая генерирует последовательность из 10 случайных чисел (в диапазоне от 1 до 10), выводит эти числа на экран и вычисляет их среднее арифметическое.

Для получения случайных чисел использовать генератор случайных чисел:

```

int r;           // случайное число
time_t t;       // текущее время - для инициализации
               // генератора случайных чисел
srand((unsigned) time(&t)); // инициализация генератора случайных чисел
...
r = rand() % 10 + 1;

```

Рекомендуемый вид экрана программы приведен ниже.

```

Случайные числа
1
3
4
2
7
4
9
6
2
1
Сред, арифм.: 3.90

```

17. Написать программу, которая выводит на экран таблицу стоимости, например, яблок в диапазоне от 100 г до 1 кг с шагом 100 г. Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным).

```

Введите цену одного килограмма и нажмите <Enter> (копейки от рублей
отделяйте точкой)
-> 16.50
Вес, гр.      Стоимость, руб.
100           1.65
200           ' 3.30
300           4.95
400           6.60
500           8.25
600           9.90
700          11.55
800          13.20
900          14.85
1000         16.50

```

18. Написать программу, которая выводит таблицу значений функции  $y = |x|$ . Диапазон изменения аргумента от -4 до 4, шаг приращения аргумента 0,5.

19. Написать программу, которая выводит таблицу значений! функции  $y = |x-2| + |x+1|$ . Диапазон изменения аргумента от -4 до 4, шаг приращения аргумента 0,5.

20. Написать программу, которая выводит на экран таблицу умножения, например, на 7. Рекомендуемый вид экрана программы приведен ниже.

```
7x2=14
7x3=21
7x4 = 28
7x5=35
7x6=42
7x7=49
7x8=56
7x9=63
```

### Задание №3 повышенной сложности

21. Написать программу проверки знания таблицы умножения. Программа должна вывести 10 примеров и выставить оценку: за 10 правильных ответов — "отлично", за 9 и 8 — "хорошо", за 7 и 6 — "удовлетворительно", за 6 и менее — "неудовлетворительно". В решение задачи использовать механизм случайных чисел см. описание в задаче 16 выше.

Ниже приведен рекомендуемый вид экрана программы. Ответы пользователя выделены полужирным.

```
*** Проверка знания таблицы умножения ***
После примера введите ответ и нажмите <Enter>.
5x3=15
7x7=49
1x4 = 4
4x3=12
9x4=36
8x8=64
7x8=52
Вы ошиблись! 7x8=56
4x7=28
3x5=15
2x5=10
Правильных ответов: 9
Оценка: Хорошо.
```

22. Написать программу проверки умения складывать и вычитать числа в пределах 100. Программа должна вывести 10 примеров, причем в каждом примере уменьшаемое должно быть больше или равно вычитаемому, т. е. не допускается предлагать испытуемому примеры с отрицательным результатом. Оценка выставляется по следующему правилу: за 10 правильных ответов — "отлично", за 9 и 8 — "хорошо", за 7 и 6 — "удовлетворительно", за 6 и менее — "плохо". В решение задачи использовать механизм случайных чисел см. описание в задаче 16 выше.

Ниже приведен рекомендуемый вид экрана программы. Ответы пользователя выделены полужирным.

```
Проверка умения складывать и вычитать числа.
После примера введите ответ и нажмите <Enter>
```



```
75-4=71
35-9=29
Вы ошиблись! 35-9=26
14-1=13
6-5=1
37-19=28
Вы ошиблись! 37-19=18
53-14=39
94-87=7
90-16=74    I
4-2 = 2
89-41=48
Правильных ответов: 8
Оценка: Хорошо
```

23. Написать программу, которая выводит на экран "электронные часы", работающие в течение, например, трех минут или до тех пор, пока пользователь не нажмет любую клавишу.

В работе использовать вызовы функций:

```
clrscr(); // очистить экран
setcursortype(_NOCURSOR); // убрать курсор
delay(1000); // задержка 1000 ms
gotoxy(1,3); // курсор в 1-ю колонку 3-й строки
kbhit() // возвращает не нулевое значение если была нажата любая
клавиша клавиатуры
_setcursortype(_NORMALCURSOR); // восстановить курсор
```

#### Индивидуальное задание №4

Реализуйте индивидуальное задание №3 с использованием цикла while или do ... while.

#### Массивы

Приступая к решению задач этого раздела, следует вспомнить что:

Массив — это структура данных, представляющая собой набор, совокупность, элементов одного типа.

В инструкции объявления массива указывается количество его элементов.

Элементы массива нумеруются с нуля. Доступ к элементу массива осуществляется путем указания индекса (номера) элемента. В качестве индекса можно использовать константу, переменную или выражение целого типа. Индекс может меняться от 0 до  $n - 1$ , где  $n$  — число элементов массива.

Доступ к элементам массива можно осуществить при помощи указателя.

В инструкции объявления массива удобно использовать именованную константу, объявленную в директиве #define. Для ввода, вывода и обработки массивов удобно использовать инструкции циклов (for, while).

Типичной ошибкой при работе с массивами является обращение к несуществующему элементу, т. е. выход индекса за допустимые пределы значений.

Рассмотрим пример задачи в решении которой используются массивы.

Написать программу, которая записывает введенные с клавиатуры данные в одномерный массив целого типа, состоящий из семи элементов. Перед вводом каждого элемента должна выводиться подсказка с номером элемента. После ввода последнего элемента программа должна вывести введенный массив и вычислить среднее арифметическое его элементов.

```
Ввод массива целых чисел
После ввода каждого числа нажмите <Enter>.
a[0] -> 10
a[1] -> 16
a[2] -> 14 .
a[3] -> 5
a[4] -> 10
a[5] -> 22
a[6] -> 22
Массив: 10 16 14 5 10 22 22
Среднее арифметическое: 14.00
```

Введите, скомпилируйте и проверьте правильность работы решения данной задачи:

**// Ввод, вывод и обработка массива**

```
#include <stdio.h>
#include <conio.h>
void main()
{
    int a[7]; // массив
    int sum;  // сумма элементов массива
    float m;  // среднее арифметическое

    int j;

    printf("\nEnter the integer array \n");
    printf("After entering of each number press <Enter>\n");

    // ввод массива
    for ( j = 0; j < 7; j++)
    {
        printf("a[%i] -> ", j);
        scanf("%i",&a[j]);
    }

    // вывод массива
    printf("\nArray is following: \n");
    for ( j = 0; j < 7; j++)
    {
        printf("%i    ", a[j]);
    }

    sum = 0;
    // вычислить сумму элементов
    for ( j = 0; j < 7; j++)
    {
        sum = sum + a[j];
    }

    m = sum / 7;

    printf("\nAverage arithmetical: %f", m);

    printf("\nFor termination press <Enter>\n");
    getch();
}
```

## Индивидуальное задание №5

1. Написать программу, которая вводит с клавиатуры данные в одномерный массив дробного типа, состоящий из пяти элементов, после чего выводит количество ненулевых элементов. Перед вводом каждого элемента должна выводиться подсказка с номером элемента.

```
Ввод массива целых чисел
После ввода каждого числа нажмите <Enter>
a[1] -> 12
a[2] -> 0
a[3] -> 3
a[4] -> -1
a[5] -> 0
```

2. Написать программу, которая выводит минимальный элемент введенного с клавиатуры массива целых чисел. Для считывания массива используйте следующую конструкцию:

```
for (i = 0; i < HB; i++) // HB символьная константа #define HB 5
scanf("%i", &a[i]);
```

3. Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным).

```
Поиск минимального элемента массива
Введите в одной строке 5 целых чисел и нажмите <Enter> >
2 0 45 -5 12
Минимальный элемент массива: -5
```

4. Написать программу, которая выводит минимальный элемент введенного с клавиатуры массива целых чисел. Для доступа к элементам массива используйте указатель. Для считывания массива используйте следующую конструкцию:

```
int a[HB]; // массив HB символьная константа #define HB 5
int *min; // номер минимального элемента
int *p; // указатель на элемент массива
p = a;
for (i = 0; i < HB; i++)
scanf("%i", p++);
```

5. Написать программу, которая вычисляет среднее арифметическое ненулевых элементов введенного с клавиатуры массива целых чисел. Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным).

```
Введите элементы массива (10 целых чисел в одной строке) и нажмите <Enter>
-> 23 0 45 -5 12 0 -2 30 0 64
■-----
-----
-----
-----:-----
.
Сумма элементов массива: 184
Количество ненулевых элементов: 7
Среднее арифметическое ненулевых элементов: 23.86
```

6. Написать программу, которая вычисляет среднее арифметическое элементов массива без учета минимального и максимального элементов массива. Ниже приведен рекомендуемый вид экрана

программы (данные, введенные пользователем, выделены полужирным).

Среднее арифметическое без учета min и max значений Введите массив (10 целых чисел в одной строке)  
-> 12 10 5 7 15 4 10 17 23 7  
Минимальный элемент: 4  
Максимальный элемент: 23  
Среднее арифм. без учета min и max значений:

7. Написать программу, которая вычисляет среднюю (за неделю) температуру воздуха. Исходные данные должны вводиться во время работы программы. Рекомендуемый вид экрана программы приведен ниже данные, введенные пользователем, выделены полужирным).

Введите температуру воздуха за неделю  
Понедельник -> 12  
Вторник -> 10  
Среда -> 16  
Четверг -> 18  
Пятница -> 17  
Суббота -> 16  
Воскресенье -> 14  
Средняя температура за неделю: 14.71 град.

8. Написать программу, которая проверяет, находится ли введенное с клавиатуры число в массиве. Массив должен вводиться в процессе работы программы.
9. Написать программу, которая проверят, представляют ли элементы введенного с клавиатуры массива возрастающую последовательность.
10. Написать программу, которая вычисляет, сколько раз введенное с клавиатуры число встречается в массиве.
11. Написать программу, которая проверяет, есть ли во введенном с клавиатуры массиве элементы с одинаковым значением.
12. Написать программу, которая методом прямого выбора сортирует по убыванию введенный с клавиатуры одномерный массив.
13. Написать программу, которая методом обмена ("пузырька") сортирует по убыванию введенный с клавиатуры одномерный массив. Метод обмена ("пузырька") заключается в прохождении массива размера N и сравнения каждого элемента с каждым раз, если элемент с индексом «к» больше элемента «к+1» то эти элементы меняют местами.
14. Написать программу, которая объединяет два упорядоченных по возрастанию массива в один, который также должен быть; упорядочен по возрастанию. Рекомендуемый вид экрана программы приведен ниже, данные, введенные пользователем, выделены полужирным.

Объединение двух массивов, упорядоченных по возрастанию  
Введите в одной строке элементы первого массива,  
( 5 целых чисел) -> 1 3 5 7 9  
Введите в одной строке элементы второго массива,  
( 5 целых чисел) -> 2 4 6 8 10  
Массив - результат  
123456789 10

Для завершения нажмите <Enter>

15. Написать программу, которая определяет количество учеников в классе, чей рост превышает средний. Рекомендуемый вид экрана программы приведен ниже. Введенные пользователем данные выделены полужирным.

```
*** Анализ роста учеников ***
Введите рост (см) учеников
Для завершения введите 0 и нажмите <Enter>
-> 175
~> 170
-> 180
-> 168,
-> 170
-> 0
Средний рост: 172.6 см
У 2 человек рост превышает средний
```

16. Написать программу, которая вводит по строкам с клавиатуры двумерный массив и вычисляет сумму его элементов по столбцам.

17. Написать программу, которая обрабатывает результаты экзамена. Для каждой оценки программа должна вычислить процент от общего количества оценок. Рекомендуемый вид экрана программы приведен ниже. Данные, введенные пользователем, выделены полужирным.

```
Обработка результатов экзамена
Введите исходные данные:
пятерок-> 12
четверок-> 10
троек-> 7
двоек-> 1
Результаты экзамена:
пятерок 12%
четверок 10%
тройка 7%
двойка 1%
Для завершения нажмите <Enter>
```

18. Написать программу, которая вводит по строкам с клавиатуры двумерный массив дробного типа (3x5 —• три строки по пять элементов) и вычисляет среднее арифметическое элементов строк. Рекомендуемый вид экрана приведен ниже.

```
Ввод и обработка массива (3x5) Введите массив по строкам
Строка 1 -> 12 15 10 22 3
Строка 2 -> 10 10 3 5 12
Строка 3 -> 11 17 10 9 7

Массив:
12 15 10 22 3
10 10 3 5 12
11 17 10 9 7
Среднее арифметическое:
Строка 1: 12.40
Строка 2: 8.00
Строка 3: 10.80
Для завершения нажмите <Enter>
```

19. Написать программу, которая определяет номер строки двумерного массива, сумма элементов которой максимальна.

### Задачи повышенной сложности

20. Написать программу, которая проверяет, является ли введенная с клавиатуры квадратная матрица "магическим квадратом". Магическим квадратом называется матрица, у которой; сумма чисел в каждом горизонтальном ряду, в каждом вертикальном и по каждой из диагоналей одна и та же (см. приведенный ниже рисунок).

2	9	4
7	5	3
6	1	8

13	8	12	1
2	11	7	14
3	10	6	15
16	5	9	4

**Рис. 2 Магический квадрат**

21. Написать программу, которая вычисляет доход по вкладу. Процентная ставка по вкладу вычисляется на основе данных, приведенных в таблице.

Сумма, тыс. руб.	Срок вклада и процентная ставка					
	3 мес	6 мес.	12 мес.	18 мес.	24 мес.	36 мес.
до 50	15,0 %	16,5 %	18,0 %	19,5 %	21,0 %	22,0 %
до 250	16,5 %	18,0 %	19,5 %	21,0 %	22,5 %	24,0 %
свыше 250	18,5 %	19,5 %	21,0 %	22,5 %	24,0 %	27,0 %

22. Написать программу, которая обрабатывает результаты спортивных соревнований, — выстраивает команды в соответствии с количеством набранных очков, вычисляемым по формуле  $K = 7N(\text{gold}) + 6N(\text{silver}) + 5N(\text{bronze})$ , где  $N_g$ ,  $N_s$  и  $N_b$  — количество золотых, серебряных и бронзовых медалей. Рекомендуемый вид экрана приведен ниже (данные, введенные пользователем, выделены полужирным).

Результат соревнований					
Австрия->	0	1	1		
Германия->	1	0	2		
Норвегия->	4	2	1		
Россия->	2	3	2		
Финляндия->	1	2	2		
Команда	Зол.	Сер.	Бр.	Всего	Очков
1 Норвегия	4	2	1	7	45
2 Россия	2	3	2	7	42
3 Финляндия	1	2	2	5	29
4 Германия	1	0	2	3	17
5 Австрия	0	1	1	2	11