

## ПРАВИЛЬНЫЕ РЕШЕНИЯ ЗАДАНИЙ

второго (заключительного) этапа Республиканской школьной олимпиады  
«Будущее Республики» по общеобразовательному предмету  
«Информатика и ИКТ», проведенного 27 февраля 2021 года  
ГОУ ВПО «Донецкий национальный технический университет»

### ЗАДАНИЯ 1-го УРОВНЯ

(максимальная оценка каждого задания – 4 балла)

Задания 1.1 – 1.4: Вычисления производятся по правилу сложения/вычитания чисел с весовыми значениями, равными 16, с учетом переносов и заемов. Можно использовать перевод в 10-ю систему, вычисления в ней и перевод обратно.

1.1. Вычислить в 16-й системе счисления:  $A17, C7_{16} + BE5, F4_{16} = 15FD, BB_{16}$

1.2. Вычислить в 8-й системе счисления:  $417, 37_8 + 1253, 24_8 = 1672, 63_8$

1.3. Вычислить в 16-й системе счисления:  $A17, C7_{16} - 2E9, F4_{16} = 72D, D3_{16}$

1.4. Вычислить в 8-й системе счисления:  $1417, 37_8 - 1253, 64_8 = 143, 53_8$

1.5. Сколько двоичных разрядов должен содержать адрес для адресации ячейки памяти, если объем памяти 1 млн. ячеек. Дайте пояснение.

Длина адреса определяется как «Большее целое от  $\log_2 M$ , где  $M$  – объем памяти. Для  $M=1000000$  составляет 20 разрядов.

1.6. Сколько различных букв специального алфавита можно закодировать при помощи 7-ми двоичных разрядов. Дайте пояснение.

При помощи  $K$  двоичных разрядов можно закодировать  $2^K$  вариантов.  $2^7 = 128$ .

#### Критерии оценивания:

Есть ход решения и правильный ответ – 4 балла.

Ход решения верен, но допущены ошибки вычислений – 1-3 балла (в зависимости от количества ошибок).

Ответ верен, но нет хода решения – 2 балла.

### ЗАДАНИЯ 2-го УРОВНЯ

(максимальная оценка каждого задания – 8 баллов)

#### Критерии оценивания.

Содержание критерия	Баллы
Учащийся выбрал правильный путь решения, из письменной записи решения понятен ход его рассуждений, получен верный ответ. Но недостаточно подробно описан ход рассуждений	7-8
В решении допущена ошибка, не имеющая принципиального характера и не влияющая на общую правильность хода решения. Выполнено правильно более половины задания.	5-6
Правильно выполнены вычисления, но ход действий не показан	4-3

Показан неполный или неверный ход действий, результат отсутствует или неверный.	1
Решение отсутствует или допущены грубые ошибки, которые свидетельствуют о незнании предложенного материала.	0
Максимальный балл	8

### Задание 2.1

Опишите работу фрагмента программы и покажите, что будет выведено на экран:

Фрагмент программы на С	Фрагмент программы на Паскале
<pre>for (i=0; i&lt;5; i++)     j=0; do {     for (k=0; k&lt;5; k++);     printf("%d", i+j+k);     j++; } while (j&lt;5);</pre>	<pre>for i:= 0 to 4 do     j:=0; repeat     for k:=0 to 4 do;         write(i+j+k);         j:=j+1;     while j&gt;4;</pre>

Ответ: Фрагмент описывает вложенные циклы. Уровень вложенности – 2.

С: Первый цикл for параметр i изменяется от 0 до 4. Тело цикла – это только оператор j=0, который просто выполнится 5 раз. После завершения цикла i=5. Внешний цикл do изменяет параметр j также от 0 до 4. Внутренний цикл изменяет параметр k от 0 до 5. Но тело этого цикла пустое, т.е. в цикле не выполняются действия, кроме увеличения параметра k. После завершения цикла k=5. Поэтому на экран будет выведено

10 11 12 13 14

! В тексте на Паскале: Первый цикл for параметр i изменяется от 0 до 4. Тело цикла – это только оператор j=0, который просто выполнится 5 раз. После завершения цикла i=5. Внешний цикл repeat выполниться один раз, т.к. j = 0 не больше 4. Внутренний цикл изменяет параметр k от 0 до 5. Но тело этого цикла пустое, т.е. в цикле не выполняются действия, кроме увеличения параметра k. После завершения цикла k=5. Поэтому на экран будет выведено 10.

### Задание 2

Напишите программу на С или Паскале по данной блок схеме и объясните, что выполняется.

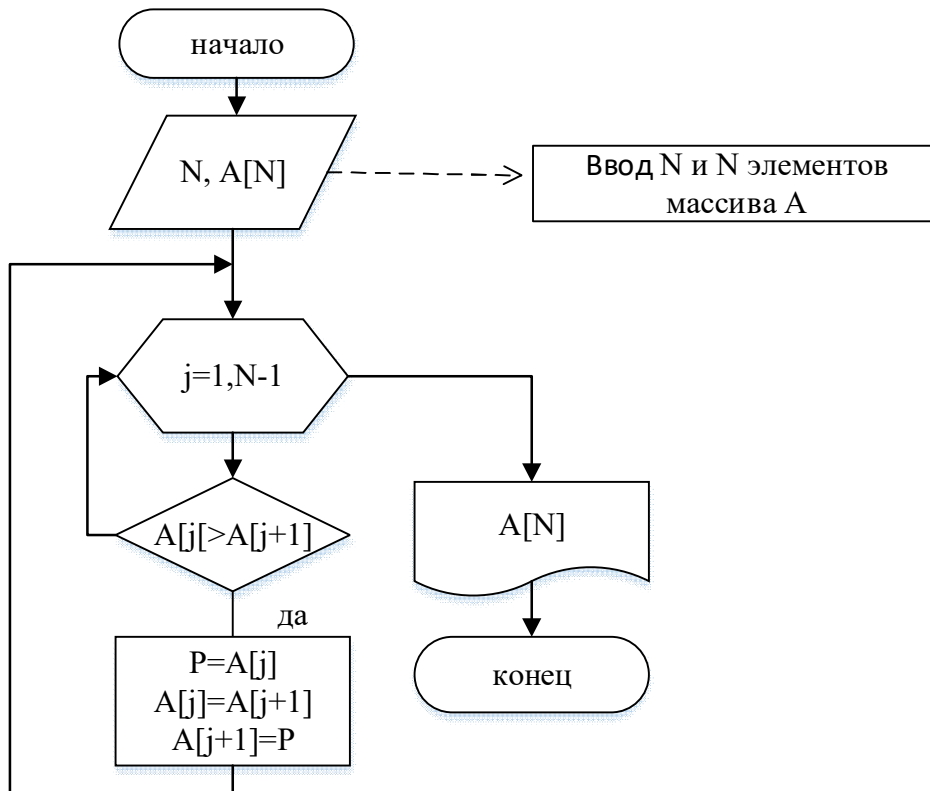
Ответ:

```
void main()
{
    int j, N;
    float A[100], P;
    do
```

```

{
printf("Введите количество элементов массива A\n"); //cout<<("Введите
количество элементов массива A\n");
scanf("%d",&N); //cin>>N;
}
while(N<3 || N>100);
for(j=0;j<N;j++)
scanf("%f",&A[j]);
for(j=0;j<N-1;j++) //j=0; while(j<N-1)
if(A[j]>A[j+1])
{
P=A[j];
A[j]=A[j+1];
A[j+1]=P;
j=-1; //j=0;
//else j++;
}
for(j=0;j<N;j++)
printf("%f ",A[j]);
}

```



В результате получим массив, отсортированный по возрастанию.

На Паскале так нельзя нужно через цикл while или repeat

### ЗАДАНИЯ 3-го УРОВНЯ

(максимальная оценка каждого задания – 12 баллов, необходимо привести пример выполнения, краткое описание алгоритма, а также программу на любом алгоритмическом языке программирования)

3.1. Целое число задано массивом цифр  $D[n]$ , где первый элемент массива содержит младшую цифру числа, а последний - старшую. Вычислить десятичное значение числа. Ограничение: количество операций умножения не должно превышать  $n$ .

Пример:  $D[5]=\{2,3,4,6,5\}$ . Ответ:  $((((5*10+6)*10+4)*10+3)*10=56432$ .

Решение 2:  $2+3*10+4*100+5*1000+5*1000$  не является оптимальным. Особенно, если 100, 1000, 1000 формируются не умножением в цикле на 10, а использованием функции «степень».

```
void main()
{
    int i,n,Num;
    int D[10];
    // ввод n, D[n]
    Num=D[n-1];
    for(i=n-2; i>=0; i--)
        Num=Num*10+D[i];
    // вывод Num
}
```

3.2. Задан массив целых чисел  $Nums[n]$ , упорядоченный по возрастанию. Ввести с клавиатуры число  $A$ . Определить индекс  $i$  элемента массива  $Nums[n]$ , значение которого равно  $A$ . Ограничение: количество операций сравнения не должно превышать  $n/2$ .

Решение: применить метод деления пополам.

Пример:  $Nums[10]=\{1,2,3,4,5,6,7,8,9,10\}$ ,  $A=6$

$il=0; in=9; im=(0+9)/2=4; Nums[4]=5 < A=6 =>$

$il=5; in=9; im=(5+9)/2=7; Nums[7]=8 > A=6 =>$

$il=5; in=7; im=(5+7)/2=6; Nums[6]=7 > A=6 =>$

$il=5; in=6; im=(5+6)/2=5; Nums[5]=6 = A=6 =>$  завершение

Элемент может не быть в массиве, следует вывод сообщения об этом. Перед циклом стоит проверить последний элемент, т.к. деление на 2 для целых чисел может не дать последний индекс.

```
void main()
{
    int A,n,Nums[100];
    int il,in,im;
    // ввод n, Nums[n], A
    il=0; in=n-1; im=(il+in)/2;
    If (Nums[n-1]==A) // вывод, что элемент A - последний в массиве
    else
    {
        while ((il<in)&&(Nums[im]!=A))
        {
            if (Nums[im]< A) in=in; else il=im;
            im=(il+in)/2;
        }
        if (Nums[im]!=A) // вывод, что элемента A нет в массиве
        else // вывод im в качестве позиции числа
    }
}
```

3.3. Задан массив чисел  $\text{Nums}[n]$ . Ввести с клавиатуры число  $A$  и вставить его в качестве элемента массива перед последним отрицательным элементом (хотя бы один отрицательный есть). Ограничение: проход по массиву (просмотр, анализ, перезапись и т.п.) должен выполняться не более 1 раза.

Решение: необходимо рассматривать элементы с конца массива и сразу их сдвигать на одну позицию, пока не будет найден отрицательный элемент. При нахождении отрицательного элемента прекратить цикл. Записать  $A$  на «бывшее место отрицательного элемента».

Пример:  $\text{Nums}[10]=\{1,-2,13,-4,5,16,-7,8,19,1\}$ ,  $A=0$ .

$i=9$   $\text{Nums}[9]=1 > 0 \Rightarrow \text{Nums}[10]=\text{Nums}[9]=1$

$i=8$   $\text{Nums}[8]=19 > 0 \Rightarrow \text{Nums}[9]=\text{Nums}[8]=19$

$i=7$   $\text{Nums}[7]=8 > 0 \Rightarrow \text{Nums}[8]=\text{Nums}[7]=8$

$i=6$   $\text{Nums}[6]=-7 < 0 \Rightarrow$  Выход из цикла;  $\text{Nums}[7]=\text{Nums}[6]=-7$ ;  $\text{Nums}[6]=A=0$ .

```
void main()
{
    int A,i,n,Nums[100];
    // ввод n, Nums[n], A
    i=n-1;
    while ((i>=0)&&( Nums[i]>=0))
    {
        Nums[i+1]= Nums[i];
        i--;
    }
    if (i>=0)
    {
        Nums[i+1]= Nums[i]; Nums[i]= A;
        // вывод массива Nums[n+1]
    }
    else //Вывод, что «массив не содержит отрицательного, это неверно»
}
```

3.4. Задан двумерный массив (прямоугольная матрица) целых чисел  $\text{Matr}[n][m]$ . В каждой строке найти первый отрицательный элемент и записать его индекс (номер столбца) в соответствующий элемент (номер элемента = номер строки) одномерного массива (вектора)  $\text{IndO}[n]$ .

Решение: Здесь, если нет отрицательного элемента, то в массив надо записать «несуществующий индекс» в качестве информации об этом, т.к. по заданию вектор результатов должен иметь столько же элементов, сколько было строк.

Пример: для нумерации элементов с 0 (язык C):

$\text{Matr}[4][5]=\{1, 2, -1, -2, 4,$       $\text{IndO}[4] = \{2, -1, 0, 3\}$

1, 2, 1, 2, 4,

-1, 2, -3, 2, 5,

5, 5, 4, -3, 2}

```
void main()
{
    int i,j,n,m,Matr[10][10],IndO[10];
    // ввод n,m, Matr[n][m]
    for(i=0; i<n; i++)
```

```

    {
        j=0;
        while((j<m)&& (Matr[i][j]>=0)) j++;
        if (j<m) IndO[i]=j; else IndO[i]=-1;
    }
//Вывод массива IndO[n]
}

```

3.5. Задан двумерный массив (квадратная матрица) целых чисел  $Matr[n][n]$ . Выполнить обмен строк и столбцов матрицы. Ограничение: минимизировать количество операций обмена элементов.

Решение: нужно рассматривать только элементы выше главной диагонали и выполнять обмен  $(i,j)$ -го элемента с  $(j,i)$ -м. Элементы, лежащие на главной диагонали, не изменяются.

```

void main()
{
    int R,i,j,n,Matr[10][10];
    // ввод n, Matr[n][n]
    for(i=0; i<n-1; i++)
        for (j=i+1; j<n; j++)
        {
            R=Matr[i][j]; Matr[i][j]=Matr[j][i]; Matr[j][i]=R;
        }
    // вывод матрицы
}

```

Пример:  $i=0; j=1,2,3,4; i=1; j=2,3,4; i=2; j=3,4; i=3; j=4;$

1	6	11	16	21	=>	1	2	3	4	5
2	7	12	17	22		6	7	8	9	10
3	8	13	18	23		11	12	13	14	15
4	9	14	19	24		16	17	18	19	20
5	10	15	20	25		21	22	23	24	25

### Критерии оценивания

Содержание ответа должно состоять из примера выполнения и программы на любом алгоритмическом языке.

Алгоритм и код программы обеспечивают полное и оптимальное решение поставленной задачи, программа снабжена необходимым количеством комментариев	12 баллов
Алгоритм и код программы обеспечивают полное и точное, но не оптимальное решение поставленной задачи	6-8 баллов
Алгоритм и код программы в целом верные, но есть ошибки при обработке входных данных, не учтены все варианты входных данных, есть ошибки при выводе данных; комментарии в программе отсутствуют.	снимается до 2-х баллов
В алгоритме или коде программы присутствуют ошибки, приводящие к неполному или неточному решению поставленной задачи; комментарии в программе отсутствуют.	снимается до 4-х баллов

Если ответ неверный, но ход рассуждения правильный	2 балла
Ответ отсутствует	0 баллов