

Методические рекомендации  
по решению задач практической части предпрофессионального экзамена  
для выпускников, обучавшихся в рамках проекта  
«Инженерный класс в московской школе», выбравших для практической части  
направление: «Информатика»

Авторы:

А.П. Лапуть, доцент кафедры ИиВПП

А.С. Максимов, профессор кафедры ИиВПП

Москва, 2020

## Оглавление

1. Задачи практической части .....	3
Тема 1. Моделирование систем (логико-структурные модели, таблицы истинности, графы).....	3
Тема 2. Системы счисления.....	4
Тема 3. Понятие энтропии, определение количества передаваемой информации .....	5
Тема 4. Решение задач на множества с использованием логических выражений (круги Эйлера).....	8
Тема 5. Анализ алгоритма, содержащего цикл и ветвление (блок-схемы и программный код) .....	11
Тема 6. Работа с массивами заполнение .....	14
Тема 7. Работа с массивами поиск по условию .....	15
Тема 8. Работа с массивами сортировка.....	16
Тема 9. Работа с массивами поиск максимума.....	17
Тема 10. Работа с массивами поиск минимума .....	18
Список использованных источников .....	20

## 1. Задачи практической части

### Тема 1. Моделирование систем (логико-структурные модели, таблицы истинности, графы)

#### Пример №1.

Дана структурная схема (рисунок 1) некоторой системы, каждый из элементов принимает значение из множества  $\{0,1\}$ . Где  $x_1, x_2, x_3, x_4, x_5$  состояния элементов системы, например 1-элемент работает, 0-элемент не работает.

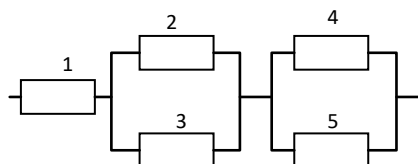


Рисунок 1.

Состояние системы также принимает значения из множества  $\{0,1\}$ .

**Найти:** Определить значение функции  $\varphi$ , при следующих значениях  $x_1, x_2, x_3, x_4, x_5$ .

$x_1$	$x_2$	$x_3$	$x_4$	$x_5$
1	1	0	1	0

#### Решение примера №1.

1) Составляем логическую функцию системы; соответствующую рабочему состоянию системы:

$$\varphi = x_1 \text{ И } (x_2 \text{ ИЛИ } x_3) \text{ И } (x_4 \text{ ИЛИ } x_5) \quad (1)$$

2) Это же выражение можно записать с использованием обозначений, принятых в математической логике:

$$\varphi = x_1 \wedge (x_2 \vee x_3) \wedge (x_4 \vee x_5) \quad (2)$$

3) Выполним арифметизацию логических операций:

Для параллельного включения элементов 2 и 3 на рисунке 1:

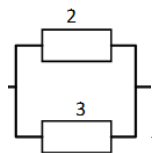


Рисунок 2

Запись  $\varphi_{23}$  будет:

$$\varphi_{23} = (1 - (1 - x_2) \cdot (1 - x_3)) \quad (3)$$

Для параллельного включения элементов 4 и 5 на рисунке 1:

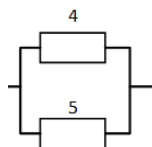


Рисунок 3

Запись  $\varphi_{45}$  будет:

$$\varphi_{45} = (1 - (1 - x_4) \cdot (1 - x_5)) \quad (4)$$

Запишем  $\varphi_{общее}$ :

$$\begin{aligned} \varphi_{общее} &= x_1 \cdot \varphi_{23} \cdot \varphi_{45} = \\ &= x_1 \cdot (1 - (1 - x_2) \cdot (1 - x_3)) \cdot (1 - (1 - x_4) \cdot (1 - x_5)) \end{aligned} \quad (5)$$

Подставляем в  $\varphi_{общее}$  значения  $x_1, x_2, x_3, x_4, x_5$ :

$$\varphi_{общее} = 1 \cdot (1 - (1 - 1) \cdot (1 - 0)) \cdot (1 - (1 - 1) \cdot (1 - 0)) = 1 \quad (6)$$

**Ответ:**  $\varphi = 1$

## Тема 2. Системы счисления

*Система счисления* — это способ представления чисел и соответствующие ему правила действия с числами.

В *позиционных системах* счисления величина, обозначаемая цифрой в записи числа, зависит от ее позиции. Количество используемых цифр называется *основанием* позиционной системы счисления. В современной математике используется позиционная десятичная система. Ее основание равно десяти, так как запись любых чисел производится с помощью десяти цифр: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. Если требуется указать основание системы, к которой относится число, то оно приписывается нижним индексом к этому числу. Например:  $101101_2$ ,  $3671_8$ ,  $3B8F_{16}$ . В системе счисления с основанием  $q$  ( $q$ -ичная система счисления) единицами разрядов служат последовательные степени числа  $q$ , т. е.  $q$  единиц какого-либо разряда образуют единицу следующего разряда. Для записи числа в  $q$ -ичной системе счисления требуется  $q$  различных знаков (цифр), изображающих числа  $0, 1, \dots, q - 1$ . Запись числа  $q$  в  $q$ -ичной системе счисления имеет вид 10. Развернутой формой записи числа называется запись в виде

$$\begin{aligned} A_q &= \pm(a_{n-1}q^{n-1} + a_{n-2}q^{n-2} + \dots + \\ &+ a_0q^0 + a_{-1}q^{-1} + a_{-2}q^{-2} + \dots + a_{-m}q^{-m}) \end{aligned} \quad (7)$$

Здесь  $Aq$  — само число,  $q$  — основание системы счисления,  $a_i$  — цифры данной системы счисления,  $n$  — число разрядов целой части числа,  $m$  — число разрядов дробной части числа.

### Пример №2

У жителей города «Недесятичное» на складе имеется 120 килограмм сырья для производства, из них 53 кг сахара и 34 кг гречневой крупы. Какая система счисления используется жителями города?

#### Решение примера №2.

Запишем уравнение, где  $X$  - основание искомой системы счисления:

$$120x = 53x + 34x \quad (8)$$

Переведем числа в десятичную систему счисления:

$$1 \cdot x^2 + 2 \cdot x^1 + 0 \cdot x^0 = 5 \cdot x^1 + 3 \cdot x^0 + 3 \cdot x^1 + 4 \cdot x^0 \quad (9)$$

После перевода чисел в десятичную систему счисления получим уравнение:

$$x^2 + 2x + 0 = 5x + 3 + 3x + 4 \quad (10)$$

После преобразования получим:

$$x^2 - 6x - 7 = 0 \quad (11)$$

Решим квадратное уравнение.

Дискриминант равен:

$$D = b^2 - 4ac = (-6)^2 - 4 \cdot 1 \cdot (-7) = 64 \quad (12)$$

Дискриминант  $D > 0$ , следовательно, уравнение имеет два действительных корня.

$$x_1 = \frac{-b + \sqrt{D}}{2a} = \frac{-(-6) + \sqrt{64}}{2 \cdot 1} = 7 \quad (13)$$

$$x_2 = \frac{-b - \sqrt{D}}{2a} = \frac{-(-6) - \sqrt{64}}{2 \cdot 1} = -1 \quad (14)$$

Отрицательное  $x_2 = -1$ ,  $x_2$  отбрасываем, положительное решение квадратного уравнения  $x_1 = 7$

**Ответ:** 7 (семеричная система счисления)

### Тема 3. Понятие энтропии, определение количества передаваемой информации

*Энтропия. Количество информации*

Свойство полноты информации предполагает, что имеется возможность измерять количество информации. Для оценки и измерения количества

информации в сообщении применяются различные подходы, среди которых следует выделить статистический, алфавитный и объемный[1].

*Количеством информации* называют ее числовую характеристику, отражающую ту степень неопределенности, которая исчезает после получения информации. Для оценки и измерения количества информации в сообщении применяются различные подходы, среди которых следует выделить статистический и алфавитный.

*Статистический подход.* Для количественной оценки неопределенности или энтропии  $H$  американский инженер Р. Хартли предложил формулу, содержащую логарифм от числа равновероятных возможностей  $N$ :

$$H = \log_2 N, \quad (15)$$

которую можно записать в следующем виде:

$$2^H = N, \quad (16)$$

где  $H$  — количество информации.

Минимальной единицей количества информации, именуемой битом, будет выбор из двух возможностей.

При не равновероятной возможности выбора количество информации  $h_i$ , зависящей от индивидуальной вероятности  $P_i$   $i$ -го выбора, вычисляется по формуле К. Шеннона

$$h_i = \log_2 \left( \frac{1}{P_i} \right) \quad (17)$$

Которую можно преобразовать к виду

$$2^{h_i} = \frac{1}{P_i} \quad (18)$$

Если события равновероятны, т.е.  $p_1 = p_2 = \dots = p_N = p$ , то формула Шеннона преобразуется в формулу Хартли (которая, таким образом, представляет собой частный случай формулы Шеннона)[4].

Связь между количеством информации и вероятностью события

$$p = \frac{1}{N} \quad (19)$$

Удобнее в качестве меры количества информации пользоваться не значением  $h_i$ , а средним значением количества информации

$$H = \sum_{i=1}^N P_i h_i = \sum_{i=1}^N P_i \log_2 \left( \frac{1}{P_i} \right) \quad (20)$$

*Алфавитный подход* позволяет определить количество текстовой информации. Количество информации, которое несет каждый символ, вычисляется по формуле

$$i = \log_2 N \quad (21)$$

где  $N$  — мощность алфавита, равная количеству символов в нем.

Текст, содержащий  $K$  символов, имеет объем информации, равный

$$I = K \cdot i \quad (22)$$

Максимальное количество слов  $L$  из  $m$  букв, которое можно составить с помощью алфавита мощностью  $N$ , определяется как

$$L = N^m \quad (23)$$

### **Пример №3.**

В магазине продавали фрукты. Продавали сливу, вишню, грушу и яблоки.

Сливы, вишни и груши продали одинаковое количество килограмм.

Вишни продали 2 килограмма.

Сообщение о том, что закончились вишня, несет 3 бита информации.

Сколько килограмм яблок продали в магазине?

### **Решение примера №3.**

Продали 2 килограмм вишни

$x$  – яблок

$$I_{\text{вишни}} = 3 \text{ бита} \quad (24)$$

$$P_{\text{вишни}} = \frac{2}{2+2+2+x} \quad (25)$$

$$\frac{1}{P} = 2^I \quad (26)$$

$$\frac{6+x}{2} = 2^3 \quad (27)$$

$$6 + x = 16 \quad (28)$$

$$x = 16 - 6 = 10 \quad (29)$$

**Ответ:** 10 килограмм яблок

#### Тема 4. Решение задач на множества с использованием логических выражений (круги Эйлера)

*Поисковый запрос* для поисковой системы в Интернете представляет собой ключевое слово или несколько ключевых слов, соединенных между собой знаками логических операций И, *ИЛИ*, НЕ.[2]

Процесс поиска в поисковой системе на основании поискового запроса аналогичен выборке данных в БД в соответствии с заданным условием отбора:

- если задано только одно ключевое слово, то производится поиск (выборка и включение в список найденного) всех web-страниц, в которых содержится данное ключевое слово;
- если ключевое слово задано с операцией НЕ, то производится поиск всех web-страниц, в которых не содержится данное ключевое слово;
- если ключевые слова связаны логической операцией И, то производится поиск web-страниц, в которых содержатся все эти ключевые слова;
- если ключевые слова связаны логической операцией ИЛИ, то производится поиск web-страниц, в которых содержится хотя бы одно ключевое слово.

#### *Ранжирование поисковых запросов*

Для разных поисковых запросов (содержащих различное число ключевых слов, связанных разными логическими операциями) количество найденных страниц будет разным. При этом важно помнить простые правила:

- *операция «И» (&, Д) сокращает* объём получаемого при поиске результата (уменьшает количество найденных сайтов), причем, чем *больше* в ней задействовано *операндов*, тем *меньше* будет объём получаемого списка найденных сайтов;
- *операция «ИЛИ» (|, V) увеличивает* объём получаемого при поиске результата (увеличивает количество найденных сайтов), причем, чем *больше* в ней задействовано *операндов*, тем *больше* будет объём получаемого списка найденных сайтов.



Запросы, состоящие из одного-единственного операнда (ключевого слова) и не содержащие логических операций можно рассматривать как запросы с операцией «ИЛИ» и с самым маленьким количеством операндов, т. е. в ранжированном списке такой запрос располагается непосредственно перед всеми остальными «ИЛИ»-запросами.

Таким образом, для поисковых запросов, включающих в себя только один вид логических операций (только «И» или только «ИЛИ») можно сразу определить их место в формируемом списке ранжирования — если в задаче требуется расположить запросы по возрастанию количества найденных документов, то:

- запросы с операцией «И» будут располагаться в начале списка, и чем больше в них задействовано операндов, тем такие запросы ближе к началу списка;

- запросы с операцией «ИЛИ» будут располагаться в конце списка, и чем больше в них задействовано операндов, тем такие запросы ближе к концу списка;

- запрос из одного-единственного ключевого слова будет располагаться в списке непосредственно перед запросами с операцией «ИЛИ».

Более сложен случай, когда поисковые запросы содержат оба типа логических операций — и «И», и «ИЛИ».

Иногда можно было получить правильный ответ просто методом исключения. Такой смешанный запрос располагается в ранжированном списке где-то посередине — между расположенным в его начале блоком запросов с «И» и расположенным в конце блоком запросов с «ИЛИ».

*Рассмотрим задачу1:*

Даны два смешанных запроса:

*(кошки & собаки) | кролики*

и

*(кошки | собаки) & кролики*

В первом случае будут найдены документы, в которых есть оба слова — «кошки» и «собаки», и к ним будут добавлены *все* документы со словом «кролики».

Во втором случае будут найдены документы, содержащие или слово «кошки», или слово «собаки», а из них будут отобраны только те документы, которые содержат также слово «кролики».

(Кошки & Собаки) | Кролики (рисунок 4)

(Кошки | Собаки) & Кролики (рисунок 5)



Рисунок 4



Рисунок 5

Можно считать, что «действие» (уменьшение или увеличение количества найденных документов) логических операций «И» и «ИЛИ» «ослабляется», когда они стоят в скобках, и «усиливается», когда эти операции расположены вне скобок. То есть, когда операция «И» стоит в скобках, а «ИЛИ» — вне скобок, будет найдено больше документов, чем когда операция «ИЛИ» стоит в скобках, а «И» — вне скобок.

#### Пример №4.

В языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ «|», а для логической операции «И» — символ «&».

В таблице приведены запросы и количество найденных по ним страниц некоторого сегмента сети Интернет.

Запрос	Найдено страниц (в тысячах)
Сириус & Вега	260
Вега & (Сириус   Арктур)	467

Какое количество страниц (в тысячах) будет найдено по запросу  
*Вега & Арктур* ?

Считается, что все запросы выполнялись практически одновременно, так что набор страниц, содержащих все искомые слова, не изменялся за время выполнения запросов.

#### Решение примера №4.

Построим диаграмму Венна для данной задачи, рисунок 6.



Рисунок 6

Количество запросов в данной области будем обозначать  $N_i$ .

Цель — найти  $N_5 + N_6$ .

При этом круг 1 соответствует Вега, круг 2 — Сириусу, круг 3 — Арктуре.

Из таблицы находим, что:

$$N_5 + N_4 = 260 \quad (30)$$

$$N_4 + N_5 + N_6 = 467 \quad (31)$$

$$N_5 = 131 \quad (32)$$

Тогда находим:

$$N_4 = 260 - N_5 = 260 - 131 = 129 \quad (33)$$

$$N_6 = 467 - N_4 - N_5 = 467 - 129 - 131 = 207 \quad (34)$$

$$N_5 + N_6 = 131 + 207 = 338 \quad (35)$$

**Ответ:** 338.

### Тема 5. Анализ алгоритма, содержащего цикл и ветвление (блок-схемы и программный код)

#### Циклы и массивы

Для многократного выполнения группы операторов служат *циклы*, причем в начале или конце цикла проверяется условие окончания цикла. В зависимости от этого существуют циклы с параметром, с предусловием и постусловием.

Для повышения производительности и качества работы каждый язык программирования имеет структурированный тип данных — *массив*. *Массивом* называется упорядоченная совокупность однотипных величин, имеющих общее имя, элементы которой различаются порядковыми номерами, именуемыми индексами

Элементы массива в памяти компьютера хранятся по соседству. Массивы различаются количеством индексов, определяющих их элементы. *Одномерный массив* (множество ящиков в один ряд) предполагает, что каждый его элемент имеет только один индекс. Количество элементов массива называют *размерностью*. При определении одномерного массива его размерность записывается в круглых скобках, рядом с его именем. Например, если сказано: «задан массив  $a(5)$ », это означает, что даны элементы  $a_1, a_2, \dots, a_5$ .

Одномерный массив можно представить как строку или столбец из  $m$  переменных.

**Обработка элементов массива (определение максимума/минимума, вычисление суммы, произведения, среднего и пр.)**

В цикле (для многомерного массива — во вложенных циклах) производится перебор элементов массива (полный или частичный — для фрагмента массива).

- *При поиске максимума/минимума* — за предполагаемый максимум/минимум берётся первый элемент массива либо константа, заведомо меньшая/большая любого элемента. Далее каждый очередной элемент массива сравнивается с предполагаемым максимумом/минимумом, и если этот элемент больше/меньше предполагаемого максимума/минимума, то значение этого элемента запоминается в качестве нового предполагаемого максимума/минимума. (Дополнительно при этом в отдельной переменной (переменных) может перезапоминаться индекс (индексы) очередного предполагаемого максимума/минимума.).

- *При вычислении суммы/произведения* — вначале переменной, выделенной для накопления суммы/произведения присваивается инициализационное значение (нуль — для суммы, единица — для произведения). Затем в цикле (либо вложенных циклах) выполняется перебор

элементов массива, и текущее значение суммы/произведения складывается/умножается на текущий элемент массива.

• При определении максимума/минимума, суммы, произведения, среднего значения элементов, удовлетворяющих заданному условию (например, только положительных) — дополнительно добавляется условный оператор с соответствующим условием, и требуемое действие (проверка и переприсваивание предполагаемого максимума/минимума, сложение, умножение) выполняется только при истинности этого условия. При вычислении среднего значения также предусматривается отдельная переменная-счётчик, которая увеличивается на 1 каждый раз, когда к сумме добавляется очередной удовлетворяющий условию элемент массива, и после вычисления суммы она делится на значение этого счётчика (количество вошедших в сумму элементов).

### Пример №5.

Запишите значение переменной  $a$  после выполнения фрагмента алгоритма рисунок 7:

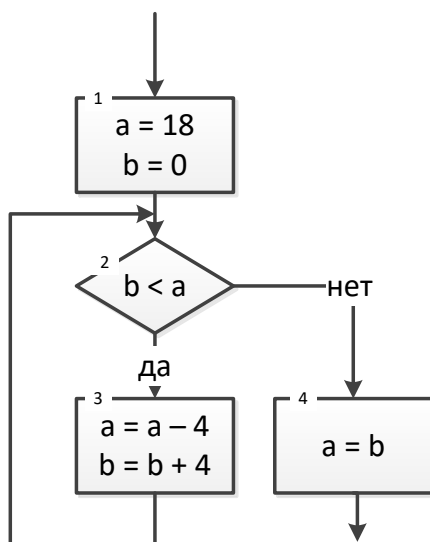


Рисунок 7

### Решение задачи №5.

На рисунке представлен циклический вычислительный процесс. Начальные значения  $a = 18$ ,  $b = 0$  (блок 1)

1) Проверяем условие  $b < a$  (блок 2), следовательно  $b=0$ ,  $a=18$ , 0 меньше 18, идем по ветке «да».

Получаем (блок 3):

$$b = b + 4 = 0 + 4 = 4 \quad (36)$$

$$a = a - 4 = 18 - 4 = 14 \quad (37)$$

$$b = 4, a=14 \quad (38)$$

2) Т.к. цикл проверяем условие  $b < a$  (блок2), соответственно  $b=4$ ,  $a=14$ ;

3) Следовательно 4 меньше 14, идем по ветке «да»

$$b = b + 4 = 4 + 4 = 8 \quad (39)$$

$$a = a - 4 = 14 - 4 = 10 \quad (40)$$

4) Затем проверяем условие  $b < a$ , соответственно  $b=8$ ,  $a=10$ ;

5) Следовательно 8 меньше 10, идем по ветке «да»

$$b = b + 4 = 8 + 4 = 12 \quad (41)$$

$$a = a - 4 = 10 - 4 = 6 \quad (42)$$

6) Затем проверяем  $b < a$  соответственно  $b=12$ ,  $a=6$

7) Следовательно 12 НЕ меньше 6, идем по ветке «нет»

8) Таким образом  $b = 12$ ,  $a = 6$ , согласно блоку 4:  $a = b$ , значит  $a = b = 12$

**Ответ:**  $a = 12$

## Тема 6. Работа с массивами заполнение

### Пример №6.

Дано два массива  $mas1$  и  $mas2$  размерностью 1 x 10. Значения элементов этих массивов задаются с помощью следующего фрагмента программы:

Сколько элементов массива  $mas2$  будут иметь отрицательные значения?

Бейсик	Паскаль	Си++
<pre>FOR i = 1 TO 10   MAS1(i) = 5 - i NEXT i FOR i = 1 TO 10   MAS2(i) = MAS1(i) + 2 NEXT i</pre>	<pre>for i := 1 to 10 do   MAS1[i] := 5 - i; for i := 1 to 10 do   MAS2[i] := MAS1[i] + 2; end;</pre>	<pre>for (i = 1; i &lt;= 10; i++) {   MAS1[i] = 5 - i; } for (i = 1; i &lt;= 10; i++) {   MAS2[i] = MAS1[i] + 2; }</pre>

### Решение задачи №6.

Расчет значений массива  $mas1$  происходит в цикле от 1 до 10 по формуле  $MAS1(i) = 5 - i$ .

Расчет значений массива  $mas2$  происходит в цикле от 1 до 10 по формуле  $MAS2(i) = MAS1(i) + 2$

Для  $i=1$  значения  $MAS1(1) = 5 - 1 = 4$  и  $MAS2(1) = MAS1(1) + 2 = 4 + 2 = 6$

Результат вычисления удобно представить в виде таблицы 1:

Таблица 1.

Массив	$mas1(1)$	$mas1(2)$	$mas1(3)$	$mas1(4)$	$mas1(5)$	$mas1(6)$	$mas1(7)$	$mas1(8)$	$mas1(9)$	$mas1(10)$
--------	-----------	-----------	-----------	-----------	-----------	-----------	-----------	-----------	-----------	------------

mas1										(10)
	4	3	2	1	0	-1	-2	-3	-4	-5

Массив mas2	mas2(1)	mas2(2)	mas2(3)	mas2(4)	mas2(5)	mas2(6)	mas2(7)	mas2(8)	mas2(9)	mas2 (10)
	6	5	4	3	2	1	0	-1	-2	-3

Таким образом, после проведения расчетов получаем, что количество отрицательных элементов в массиве MAS2 равно трем.

**Ответ:** 3 элемента массива mas2 будут иметь отрицательное значение

## Тема 7. Работа с массивами поиск по условию

### Пример №7.

Определите, что будет напечатано в результате работы следующего фрагмента программы.

Дан фрагмент программного кода. В программе дан одномерный массив А с размерностью 1 x 10. Значения элементов массива равны:

7; 17; -10; 10; 5; -1; 11; 12; -2; -2 соответственно, т. е.  $A(1) = 7$ ;  $A(2) = 17$  и т. д.

Определите значение переменной S после выполнения следующего фрагмента программы, записанного ниже:

Бейсик	Паскаль	C++
<pre>S = 0 FOR i = 1 TO 10   IF A(i) &lt; 0 THEN     S = S + A(i)   END IF NEXT i PRINT S</pre>	<pre>s := 0; for i := 1 to 10 do begin   if A[i] &lt; 0 then     S := S + A[i];   end end; writeln S;</pre>	<pre>S = 0; for (i = 1; k &lt;= 10; k++) {   if (A[i] &lt; 0 {     S=S+A[i];   } } cout&lt;&lt;"S="&lt;&lt;S;</pre>

### Решение примера №7.

В фрагменте программы записан цикл от 1 до 10, где суммируются элементы массива  $A(i)$  ( $S = S + A(i)$ ), которые меньше нуля ( $A(i) < 0$ ).

Значения элементов массива равны:

7; 17; -10; 10; 5; -1; 11; 12; -2; -2 соответственно, т. е.  $A(1) = 7$ ;  $A(2) = 17$  и т. д.

Запишем массив в виде таблицы 2.

Таблица 2.

Массив А	A(1)	A(2)	A(3)	A(4)	A(5)	A(6)	A(7)	A(8)	A(9)	A(10)
	7	17	-10	10	5	-1	11	12	-2	-2

Соответственно складываются отрицательные элементы массива А:

$$S = S + A(i) = 0 + (-10) + (-1) + (-2) + (-2) = -15$$

**Ответ:** -15

## Тема 8. Работа с массивами сортировка

### Пример № 8.

Дан одномерный массив А, количеством элементов от 1 до 10.

Элементы массива равны: 6; 7; 3; 8; 5; 1; 2; 4; 9; 2

A(1)=6; A(2)=7 и т.д.

Определите, чему будет равняться k в результате работы следующего фрагмента программы:

Бейсик	Паскаль	Си++
<pre>k = 0 FOR i = 2 TO 10   IF A(i) &lt; A(1) THEN     k = k + 1     d = A(i)     A(i) = A(1)     A(1) = d   ENDIF NEXT i PRINT k</pre>	<pre>k := 0; for i := 2 to 10 do   if A[i] &lt; A[1] then     begin       k := k + 1;       d := A[i];       A[i] := A[1];       A[1] := d;     end; writeln k;</pre>	<pre>k = 0; for (i = 2; i &lt;= 10; i++)   if (A[i] &lt; A[1])   {     k++;     d = A[i];     A[i] = A[1];     A[1] = d;   } cout &lt;&lt; "k=" &lt;&lt; k;</pre>

### Решение примера №8.

Промоделируем работу программы и запишем массив в виде таблицы 3:

Таблица 3.

Массив А до перестановки

6	7	3	8	5	1	2	4	9	2
При i	k	a(i)	a(1)						
2									



3	1	3	6
4			
5			
6	2	1	3
7			
8			
9			
10			

**Ответ: k = 2**

## Тема 9. Работа с массивами поиск максимума

### Пример №9.

Дан одномерный массив A, количеством элементов от 1 до 10.

Элементы массива равны: 0; 8; 7; 2; 5; 3; 3; -2, соответственно  $A(1) = 0$ ;  $A(2) = 8$  и т.д.

Определите, чему будет равняться  $kk$  и  $d$  в результате работы следующего фрагмента программы:

Бейсик	Паскаль	Си++
<pre>n = 8 d = 0 kk = -1000 For i = 1 To n   IF a(i) &gt;= kk THEN     kk = a(i)     d = i   END IF NEXT i PRINT "kk=";kk PRINT "d=";d</pre>	<pre>n := 8; d := 0; kk := -1000; For i = 1 To n begin   If a(i) &gt;= kk Then     kk = a(i);     d = i;   end if; end; writeln kk; writeln d;</pre>	<pre>int i, n, kk, d; d = 0; n = 8; int a[i]; kk = -1000; for (i = 1; i &lt;=n; i = ++)</pre> <pre>{   if (a[i] &gt;= kk)   { kk = a[i];     d = i;   } }</pre> <pre>cout&lt;&lt;"kk="&lt;&lt;kk; cout&lt;&lt;"d="&lt;&lt;d;</pre>

### Решение примера №9.

Фрагмент программы находит максимальный элемент в массиве A ( $a(i) \geq kk$ ) и его индекс ( $d = i$ ).

Запишем массив A в таблице 4.

Таблица 4.

Массив А							
A(1)	A(2)	A(3)	A(4)	A(5)	A(6)	A(7)	A(8)
0	8	7	2	5	3	3	-2

**Ответ:**  $kk = 8, d = 2$

### Тема 10. Работа с массивами поиск минимума

#### Пример №10.

Дан одномерный массив В, количеством элементов от 1 до 8.

Элементы массива равны: 3; 1; -7; 0; -7; -10; -11; -6, соответственно

$B(1) = 3; B(2) = 1$  и т.д.

Определите, чему будет равняться  $f5$  и  $m$  в результате работы следующего фрагмента программы:

Бейсик	Паскаль	Си++
<pre>n = 8 m = 0 f5 = 100 FOR i = 1 TO n   IF b(i) &lt;= f5 THEN     f5 = b(i)     m = i   END IF NEXT i PRINT " f5="; f5 PRINT "m="; m</pre>	<pre>n := 8; m := 0; f5 := 100; For i = 1 To n begin   If b(i) &lt;= f5 Then     f5 = b(i);     m = i;   end if; end; writeln f5; writeln m;</pre>	<pre>int i, n, f5, m; m = 0; n = 8; int b[i]; f5 = 100; for (i = 1; i &lt;=n; i = ++ ) {   if (b[i] &lt;= f5)   { f5 = b[i];     m = i;   } } cout&lt;&lt;" f5="&lt;&lt; f5; cout&lt;&lt;"m="&lt;&lt;m;</pre>

#### Решение примера №10.

Фрагмент программы находит минимальный элемент в массиве В

$(b(i) \leq f5)$  и его индекс ( $m = i$ ).

Запишем массив в таблице 5.

Таблица 5.

Массив В							
B(1)	B(2)	B(3)	B(4)	B(5)	B(6)	B(7)	B(8)
3	1	-7	0	-7	-10	-11	-6

**Ответ:**  $f5 = -11, m = 7$



### **Список использованных источников**

1) Кудинов Ю. И., Пащенко Ф. Ф., Келина А. Ю. Практикум по основам современной информатики: Учебное пособие. — СПб.: Издательство «Лань», 2011. — 352 с.: ил.

2) Богомолова, Ольга Борисовна Информатика : ЕГЭ за 30 дней : экспресс-репетитор / О. Б. Богомолова. — Москва: АСТ, Астрель, 2014. — 446, [2] с. — (Единый государственный экзамен).