

# Занятие №16

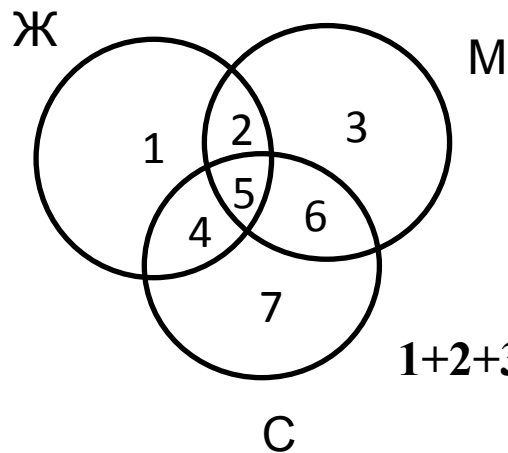
Разбор №19

В таблице приведены запросы и количество страниц, которые нашел поисковый сервер по этим запросам в некотором сегменте Интернета:

Какое наибольшее количество страниц (в тысячах) может быть найдено по запросу

**Жираф | Слон | Моська**

Запрос	Количество страниц (тыс.)
жираф	159
мостька	108
жираф   слон	195
слон & моська	54
жираф & слон & моська	40



$$1+2+3+4+5+6+7 = ?$$

жираф	$1 + 2 + 4 + 5 = 159$
мостька	$2 + 3 + 5 + 6 = 108$
жираф   слон	$1+2+4+5+6+7 = 195$
слон & моська	$5+6 = 54$
жираф & слон & моська	$5 = 40$

$1+2+4+5+6+7 = 195$ , то нам не хватает только 3 области. Она упоминается в запросе «мостька»

$2+3+5+6 = 108$  отсюда  $N3 = 108 - 54 - N2 = 54 - N2$ .

Раз нас просят найти максимальное значение суммы, а 195 у нас уже есть как константа, значит меняться значение может только из-за третьей области  $N3$ , которую мы должны максимизировать.

$N3 = 54 - N2$ . Значит максимальное значение при  $N2 = 0$  будет равно 54.

$195 + 54 = 249$ .

Ответ: 249

В таблице приведены запросы и количество страниц, которые нашел поисковый сервер по этим запросам в некотором сегменте Интернета: Какое наименьшее количество страниц (в тысячах) может быть найдено по запросу

**Наковальня**

<i>Запрос</i>	<i>Количество страниц (тыс.)</i>
<i>ухо   подкова   наковальня</i>	<i>249</i>
<i>ухо   подкова</i>	<i>227</i>
<i>ухо &amp; подкова</i>	<i>96</i>
<i>подкова &amp; наковальня</i>	<i>96</i>
<i>ухо &amp; подкова &amp; наковальня</i>	<i>48</i>

# 19 (повышенный уровень, время – 5 мин)

- массив - это набор однотипных элементов, имеющих общее имя и расположенных в памяти рядом
- для обращения к элементу массива используют квадратные скобки, запись  $\mathbf{A[i]}$  обозначает элемент массива  $\mathbf{A}$  с номером (индексом)  $\mathbf{i}$
- матрица (двухмерный массив) - это прямоугольная таблица однотипных элементов
- если матрица имеет имя  $\mathbf{A}$ , то обращение  $\mathbf{A[i, k]}$  обозначает элемент, расположенный на пересечении строки  $\mathbf{i}$  и столбца  $\mathbf{k}$
- элементы, у которых номера строки и столбца совпадают, расположены на главной диагонали<sup>1</sup>

A [1,1]			
	A [2,2]		
		A [3,3]	
			A [4,4]

- выше главной диагонали расположены элементы, у которых номер строки **меньше** номера столбца

	A [1,2]	A [1,3]	A [1,4]
		A [2,3]	A [2,4]
			A [3,4]

- ниже главной диагонали расположены элементы, у которых номер строки **больше** номера столбца

A [2,1]			
A [3,1]	A [3,2]		
A [4,1]	A [4,2]	A [4,3]	

**P-07.** В программе описан одномерный целочисленный массив с индексами от 0 до 10. Ниже представлен фрагмент программы, обрабатывающей данный массив:

```
s:=0;  
n:=10;  
for i:=0 to n-2 do begin  
    s:=s+A[i]-A[i+2]  
end;
```

В начале выполнения этого фрагмента в массиве находились трёхзначные натуральные числа. Какое наибольшее значение может иметь переменная *s* после выполнения данной программы?

**Решение:**

- 1) сначала попытаемся понять, что же делает эта программа; возьмем массив из пяти элементов

( $n = 4$ ):

0      1      2      3      4

A[0]	A[1]	A[2]	A[3]	A[4]
------	------	------	------	------

- 2) переменная *s* будет изменяться следующим образом:

**s := 0**

**s := s + A[0] - A[2]**

**s := s + A[1] - A[3]**

**s := s + A[2] - A[4]**

- 3) в итоге после всех действий

**s := A[0] - A[2] + A[1] - A[3] + A[2] - A[4] = A[0] + A[1] - A[3] - A[4]**

- 4) это значит, что значение *s* всегда будет равно сумме двух первых элементов массива минус сумма двух последних элементов

- 5) все числа – трёхзначные, то есть принадлежат отрезку [100;999]

- 6) максимальное значение *s* равно  $999 + 999 - 100 - 100 = 1798$

- 7) обратите внимание, что это число не зависит от размера массива

- 8) ответ: **1798**.

P-06. В программе используется одномерный целочисленный массив A с индексами от 0 до 9. Значения элементов равны

6; 9; 7; 2; 1; 5; 0; 3; 4; 8 соответственно, т.е.  $A[0] = 6$ ;  $A[1] = 9$  и т.д. Определите значение переменной c после выполнения следующего фрагмента программы, записанного ниже на разных языках программирования.

```
c := 0;
for i := 1 to 9 do
  if A[i-1] < A[i] then begin
    c := c + 1;
    t := A[i];
    A[i] := A[i-1];
    A[i-1] := t
  end;
```

Решение:

- 1) сначала попытаемся понять, что же делает эта программа:
  - в цикле рассматриваются пары соседних элементов, начиная с пары  $(A[0], A[1])$  и заканчивая парой  $(A[8], A[9])$ ;
  - если предыдущий элемент  $A[i-1]$  меньше следующего  $(A[i])$ , они меняются местами через вспомогательную переменную t; таким образом, цикл выполняет один этап (один проход по массиву) метода сортировки массива **по убыванию**, который называется «методом пузырька»
  - начальное значение переменной c, которая нас интересует, равно нулю; при каждой перестановке оно увеличивается на 1 (начиная с нуля), то есть, c – счётчик перестановок
- 2) для первой пары выполняется условие  $A[0] < A[1]$ , поэтому выполняется перестановка:

```
6 9 7 2 1 5 0 3 4 8
9 6 7 2 1 5 0 3 4 8
```

следующая перестановка будет для пары  $(A[1], A[2])$ :

```
9 6 7 2 1 5 0 3 4 8
9 7 6 2 1 5 0 3 4 8
```

всего было сделано 6 перестановок, при каждой счётчик увеличивался на 1, поэтому после выполнения этого фрагмента значение переменной c будет равно 6

ответ: 6.

**P-05.** В программе используется одномерный целочисленный массив  $A$  с индексами от 1 до 25. Ниже представлен фрагмент программы, в котором задаются значения элементов:

```
n := 25;
```

```
A[1] := 2;
```

```
for i := 2 to n do begin
```

```
  A[i] := 2*A[i-1] mod 10;
```

```
end;
```

Чему будет равно значение  $A[25]$  после выполнения фрагмента программы?

**Решение:**

- 1) заметим особенность: внутри цикла берется остаток от деления  $2 \cdot A[i-1]$  на 10, то есть последняя цифра десятичной записи; поэтому все элементы массива – однозначные числа
- 2) если бы не было этого взятия остатка, каждое последующее число в 2 раза больше предыдущего, цепочка начинается с 2, поэтому в массиве были бы записаны степени числа 2: 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024 и т.д.
- 3) выделим последние цифры в этой цепочке:  
**2, 4, 8, 6**, 2, 4, 8, 6, 2, 4, ...  
они повторяются через 4 элемента
- 4) таких полных групп в массиве с 25 элементами будет  $25 \operatorname{div} 4 = 6$ ; эти 6 групп займут первые 24 элемента, а 25-м будет первый элемент в четвёрке, то есть 2
- 5) Ответ: **2**.

**P-01.** Значения элементов двумерного массива  $A[1..10,1..10]$  задаются с помощью следующего фрагмента программы:

```
for i:=1 to 10 do
for k:=1 to 10 do
if i > k then
    A[i,k] := 1
else A[i,k] := 0;
```

Чему равна сумма элементов массива после выполнения этого фрагмента программы?

- 1) в программе есть вложенный цикл, в котором переменная  $i$  обозначает строку, а  $k$  – столбец матрицы
- 2) элементы, для которых  $i=k$  – это главная диагональ матрицы, поэтому элементы, для которых  $i > k$  (только они будут равны 1), находятся под главной диагональю
- 3) в первой строке единичных элементов нет, во второй есть один такой элемент, в третьей – 2, в последней (10-ой) их 9, поэтому сумма элементов массива равна

$$1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 = 45$$

таким образом, правильный ответ – 45.



**P-01.** Значения элементов двумерного массива  $A[1..10,1..10]$  задаются с помощью следующего фрагмента программы:

```
for i:=1 to 10 do
for k:=1 to 10 do
if i > k then
    A[i,k] := 1
else A[i,k] := 0;
```

Чему равна сумма элементов массива после выполнения этого фрагмента программы?

при большом размере массива (например, 100 на 100) суммирование может оказаться трудоемким, поэтому лучше вспомнить формулу для вычисления суммы элементов арифметической прогрессии (именно такая прогрессия у нас, с шагом 1):

$$S = N \cdot \frac{a_1 + a_N}{2},$$

где  $N$  - количество элементов, а  $a_1$  и  $a_N$  - соответственно первый и последний элементы последовательности; в данном случае имеем

$$S = 9 \cdot \frac{1+9}{2} = 45.$$

если приведенная выше формула прочно забыта, можно попытаться сгруппировать слагаемые в пары с равной суммой (как сделал, будучи школьником, великий математик К.Ф. Гаусс), например:

$$1 + 2 + \dots + 9 = (1+9) + (2+8) + \dots + (4+6) + 5 = 4 \cdot 10 + 5$$

24) Значения элементов двумерного массива  $A[1..100,1..100]$  задаются с помощью следующего фрагмента программы:

```
for i:=1 to 100 do
for k:=1 to 100 do
if i > k then
    A[i,k] := 1
else A[i,k] := -1;
```

Чему равна сумма элементов массива после выполнения этого фрагмента программы?

25) Значения элементов двумерного массива  $A[1..100,1..100]$  задаются с помощью следующего фрагмента программы:

```
for i:=1 to 100 do
for k:=1 to 100 do
if i > k then
    A[i,k] := i
else A[i,k] := -k;
```

Чему равна сумма элементов массива после выполнения этого фрагмента программы?

Ответ: -100; -5050

105) В программе используется одномерный целочисленный массив A с индексами от 0 до 9. Значения элементов равны 1, 2, 4, 3, 5, 6, 8, 7, 3, 4, т.е.  $A[0]=1$ ,  $A[1]=2$  и т.д. Определите значение переменной s после выполнения следующего фрагмента программы:

```
s := 0;
for i:=1 to 9 do
  if A[i-1] < A[i] then begin
    A[i] := A[i-1] + 1;
    s := s + A[i]
  end
```

100) В программе используется одномерный целочисленный массив A с индексами от 0 до 9. Значения элементов равны 2, 7, 3, 12, 6, 10, 13, 8, 3, 5, т.е.  $A[0]=2$ ,  $A[1]=7$  и т.д. Определите значение переменной s после выполнения следующего фрагмента программы:

```
n:= 10;
s:= 0;
for i:=1 to n-1 do
  if A[i] > A[i-1] then begin
    A[i]:= 2*A[i] - A[i-1];
    s:= s + A[i]
  end;
```

92) В программе используется одномерный целочисленный массив A с индексами от 0 до 9. Значения элементов равны **4, 7, 3, 8, 5, 0, 1, 2, 9, 6** соответственно, т.е.  $A[0] = 4$ ,  $A[1] = 7$  и т.д. Определите значение переменной c после выполнения следующего фрагмента этой программы:

```
c := 0;
for i := 1 to 9 do
  if A[i] < A[0] then begin
    c := c + 1;
    t := A[i];
    A[i] := A[0];
    A[0] := t;
  end;
```

85) В программе используется одномерный целочисленный массив A с индексами от 0 до 9. Значения элементов равны **7; 5; 3; 4; 8; 8; 9; 7; 6; 2** соответственно, т.е.  $A[0] = 7$ ;  $A[1] = 5$  и т.д. Определите значение переменной c после выполнения следующего фрагмента программы:

```
c := 0;
for i := 1 to 9 do
  if A[i-1] < A[i] then begin
    t:= A[i];
    A[i]:= A[i - 1];
    A[i-1]:= t
  end
else
  c:= c + 1;
```