

```

1 Program dvp_02;
2 {
    Задача. Определить матрицу A (200 * 2, 200), такую что: в 1 и 2
        строке по 200 чисел, в 3-ей и 4-ой строке по 199 чисел,
        в 5-ой и 6-ой строке - по 198 чисел и так далее.
        Элементы матрицы заполняются последовательными
        натуральными числами. Первое число первой строки - 1.
        Первое число второй строки 201, третьей - 401 и так
        далее.
        Памяти выделяется ровно столько, сколько нужно.
        С помощью MemAvail пронаблюдать за расходом памяти. После
        Begin, после выделения памяти, после ее освобождения.
        По номеру строки и столбца, узнать какой там находится
        элемент-число.
        Памяти использовать как можно меньше.

    Решение. Святослав Мельничук,
        11 класс, лицей 1,
        08.06.09, 16:47
}

20 Uses Crt;

21 Const
22     N = 10; { половина числа строк массива }
23     MAX = N; { длина верхней строки }

24 Type
25     massiv = array [1..1] of word;
26     B = ^massiv;

27 Var
28     i, { номер строки }
29     j, { номер столбца }
30     d, { длина строки }
31     k, { для проверки индексов }
32     e { число элемент матрицы }
33     : word;

34     A { массив адресов строк }
35     : array [1..N * 2] of B;

36 Begin
37     ClrScr;
38     WriteLn ('Свободно памяти в начале программы: ', MemAvail);
39     WriteLn ('Нажмите на Enter');
40     ReadLn;

41     {
        Выделим память под массив

```

```

}
44 i := 1;
45 d := MAX;
46 While i <= N * 2 do
47     begin
48         GetMem (A [i], d * SizeOf (word));
49         GetMem (A [i + 1], d * SizeOf (word));
50         Dec (d);
51         i := i + 2;
52     end;

53 WriteLn ('Память под массив выделена!');
54 WriteLn ('Осталось памяти: ', MemAvail);
55 WriteLn ('Нажмите на Enter');
56 ReadLn;

57 {
    заполним треугольный, зубчатый массив числами
}
60 i := 1;
61 e := 1;
62 d := MAX;
63 While i <= N * 2 do
64     begin
65         j := 1;
66         While j <= d do
67             begin
68                 A [i]^ [j] := e;
69                 A [i + 1] ^ [j] := e + d;
70                 Inc (j);
71                 Inc (e);
72             end;
73         e := e + d;
74         Dec (d);

75         i := i + 2;
76     end;

77 {
    напечатаем треугольный массив строками
}
80 i := 1;
81 d := MAX;
82 While i <= N * 2 do
83     begin
84         j := 1;
85         while j <= d do
86             begin
87                 Write (A [i]^ [j]:3, ' ');
88                 Inc (j);
89             end;

```

```

90     WriteLn;
91     j := 1;
92     While j <= d do
93     begin
94         write (A [i + 1]^ [j]:3, ' ');
95         Inc (j);
96     end;
97     WriteLn;

98     Dec (d);
99     i := i + 2;
100    end;

101    WriteLn;
102    Write ('Номер строки: ');
103    ReadLn (i);

104    Write ('Номер столбца: ');
105    ReadLn (j);
106    {
        защита от неверного ввода индексов матрицы
    }
107    if (i > N * 2)
108    or
109    (i <= 0)
110    then
111        WriteLn ('С такими индексами элемента не существует!')
112    else
113        begin
114            {
                найдем k - что надо вычесть из N, чтобы узнать длину
                строки
            }
115            if i mod 2 = 0
116            then
117                k := i - 1
118            else
119                k := i;
120            k := k div 2;
121            k := N - k;

122            if (j > k)
123            or
124            (j <= 0)
125            then
126                WriteLn ('С такими индексами элемента не существует!')
127            else
128                Writeln ('A [' , i, ' , ' , j, ' ] = ' , A [i]^ [j]);
129        end;

130    {

```

```

        Освободим память
    }
138     i := 1;
139     d := MAX;
140     While i <= N * 2 do
141         begin
142             FreeMem (A [i], d * SizeOf (word));
143             FreeMem (A [i + 1], d * SizeOf (word));
144             Dec (d);
145             i := i + 2;
146         end;

147     WriteLn;
148     WriteLn ('Память освобождена!');
149     WriteLn ('Осталось памяти в конце программы: ', MemAvail);

150     ReadLn;
151     End.

```

Sergey Mitrofanov, 29.09.13, 21:41

E-mail: infostar@mail.ru

© <http://www.Best-Listing.ru>, 2006–2013