

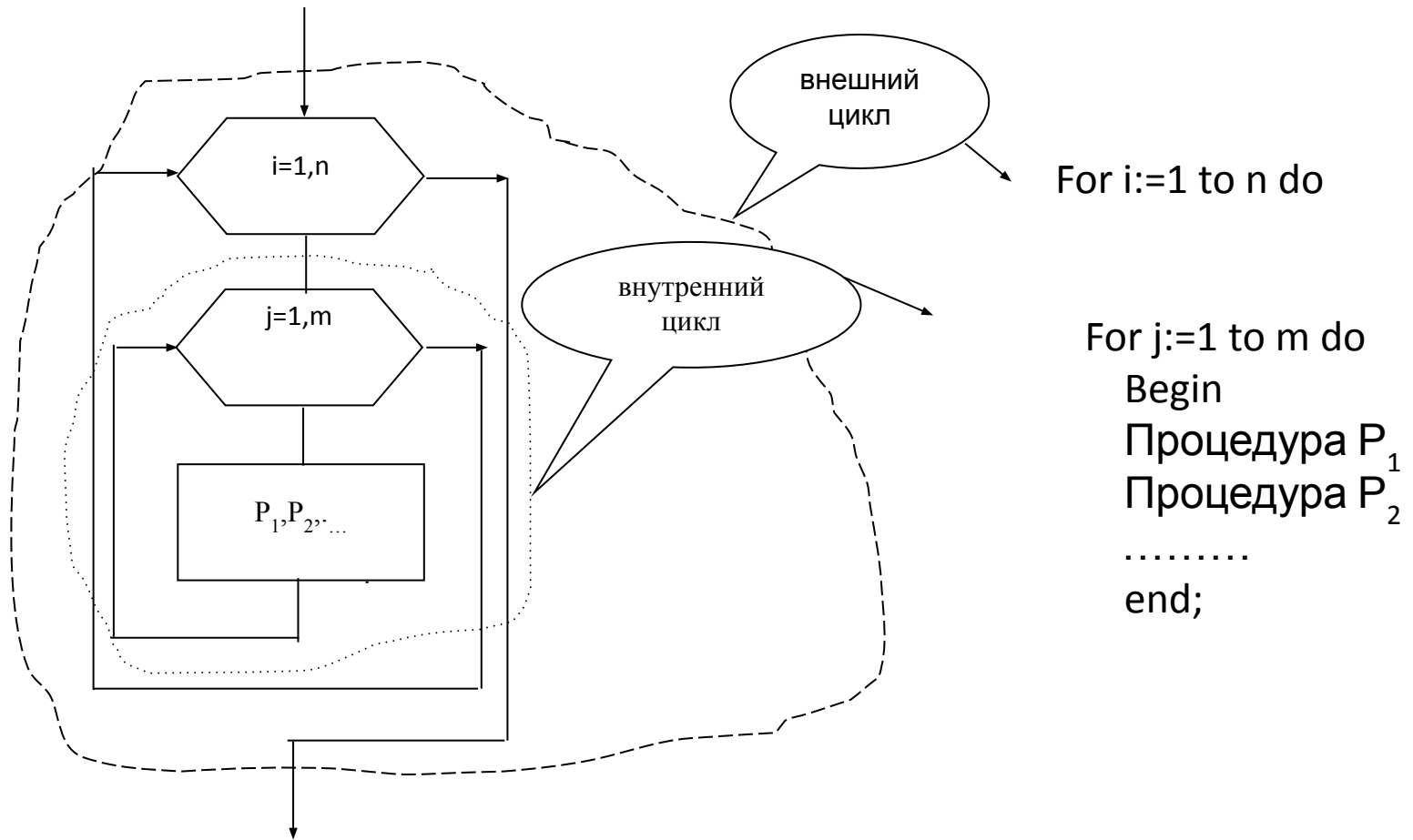
Разработка ПАСКАЛЬ - программы со сложным ЦИКЛОМ

Лабораторная работа №7

Сложные циклы

- Сложными называются такие циклические вычислительные процессы, которые состоят из двух и более включенных друг в друга циклов.
- При этом циклы, объемлющие другие циклы, принято называть **внешними**, а циклы, входящие во внешние циклы, **внутренними** (*вложенными*).

Схема алгоритма сложного цикла



Двухмерный массив

$$\left\{ \begin{array}{ccc} Y_{11} & Y_{12} & Y_{13} \\ & Y_{21} & Y_{22} & Y_{23} \end{array} \right\}$$

Описание массива

VAR

Y: ARRAY [1..20, 1..10] OF REAL;

Например:

y[1,3], y[2,5], y[10,10]

- соответствует индексированным переменным

Y_{1,3}, Y_{2,5}, Y_{10,10}

y[k,l]

Матрица чисел X



при $i=2$ и $j=1$

Разработка программы со сложным ЦИКЛОМ

- **Цель работы:** получение практических навыков по использованию вложенных друг в друга структур повторения при алгоритмизации и программировании, по разработке и использованию подпрограмм с такими структурами.
- **Задание по лабораторной работе**

Разработать подпрограмму обработки матрицы чисел, представленной в виде двумерного массива, для задачи:

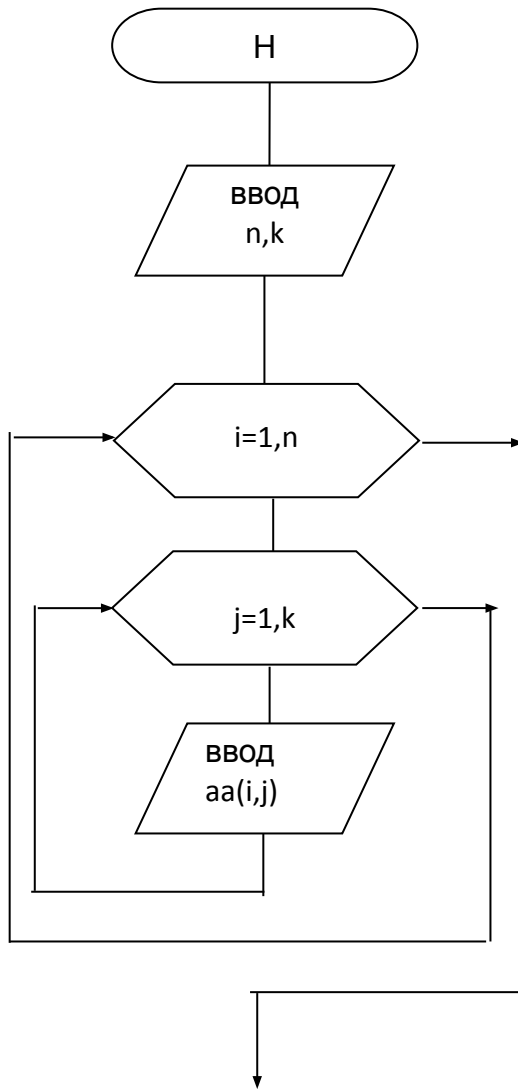
1) определения номеров строк матрицы, последние элементы которых равны сумме предыдущих элементов

Алгоритм

Для исполнения подпрограммы использовать модуль Program, в котором должны предусматриваться:

- создание двумерного массива чисел (ввод чисел в качестве элементов массива);
- вызов подпрограммы обработки массива;
- вывод результатов обработки массива (предусмотреть вариант отсутствия данных по условию задачи)

Процедура ввода



```
Var aa: ARRAY [1..10, 1..6] of real;
```

```
n,k,i,j: integer;
```

```
Begin
```

```
writeln ('Введите размеры массива - КОЛ-ВО  
строк и кол-во столбцов');
```

```
readln(n,k);
```

```
writeln('Введите элементы массива ');
```

```
for i:=1 to n do
```

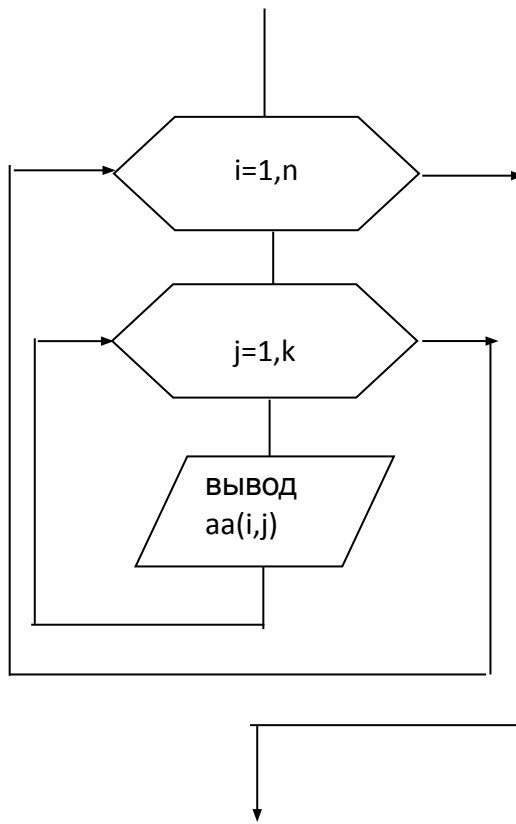
```
for j:=1 to k do read (aa[i,j]);
```

```
{ или for j:=1 to k do readln(aa[i,j]);}
```

```
End;
```

Процедура вывода

```
writeln ('Исходные данные');  
for i:=1 to n do  
  begin  
    for j:=1 to k do write (aa[i,j]:7:2);  
    writeln;  
  end;
```



Обработка двумерного массива

Вычисления сумм элементов каждой строки

```
Type ta2 = array [1..20, 1..20] of integer;
    ta1 = array [1..20] of integer;
Var  x: ta2;
     s: ta1;
     m, n, i, j: integer;
...
For i := 1 to m do
begin
    s[i] := 0;
    For j := 1 to n do s[i] := s[i] + x[i, j];
end;
...
```

Вычисления сумм элементов каждого столбца

```
Type ta2 = array [1..20, 1..20] of integer;
    ta1 = array [1..20] of integer;
Var  x: ta2;
     s: ta1;
     m, n, i, j: integer;
...
For j := 1 to n do
begin
    s[j] := 0;
    For i := 1 to m do s[j] := s[j] + x[i, j];
end;
...
```

Процедура определения номеров строк матрицы, все элементы которых принадлежат заданному диапазону

(условие: $d1 \leq x[i,j] \leq d2$)

```
Program P7;
```

```
Type ta1= array [1..20] of integer;
```

```
    ta2= array [1..20, 1..20] of integer;
```

```
Procedure PMatrix
```

```
    (x: ta2; m, n: integer; d1, d2: integer; Var num: ta1; Var k: integer);
```

```
    Var i, j: integer; r: boolean;
```

```
Begin
```

```
{Формирование массива номеров искоемых строк}
```

```
k:=0;
```

```
For i:=1 to m do
```

```
    begin
```

```
        {Поиск в строке элемента, не принадлежащего  
диапазону}
```

```
        r:= true; j:=1;
```

```
        While (j<=n) and (r= true) do
```

```
            If (x[i, j] < d1) or (x[i, j] > d2) then r:= false else j:=j+1;
```

```
            If r= true then {ВСЕ ЭЛЕМЕНТЫ В ДИАПАЗОНЕ}
```

```
                begin
```

```
                    k:=k+1; num[k]:=i;
```

```
                end;
```

```
        end;
```