**Министерство науки и высшего образования Российской Федерации**

федеральное государственное автономное образовательное учреждение

высшего образования

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**

**ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Инженерная школа природных ресурсов

Направление подготовки Химическая технология

Отделение химической инженерии

**СОСТАВЛЕНИЕ ПРОГРАММ С ИСПОЛЬЗОВАНИЕМ МАССИВОВ КАК ФОРМАЛЬНЫХ ПАРАМЕТРОВ ПОДПРОГРАММ**

**Лабораторная работа по дисциплине «Углубленный курс информатики»**

Выполнил студент гр. 2Д91 Г. Н. Омельченко

(Подпись)

\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ 2020 г.

Отчет принят:

Преподаватель

доцент ОХИ ИШПР, к.т.н. В.А. Чузлов

(Подпись)

\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ 2020 г.

Томск 2020 г.

**Цель работы:** рассмотреть и составить программы с использованием массивов как формальных параметров подпрограмм.

**Теоретическая часть**

Параметры, указываемые при описании подпрограммы, называются формальными. Параметры, указываемые при вызове подпрограммы, называются фактическими.

В языке Паскаль выделяют два вида подпрограмм:

* процедура (procedure);
* функция (function).

Функция отличается от процедуры тем, что результат её работы возвращается в виде значения этой функции, и, следовательно, вызов функции может использоваться наряду с другими операндами в выражениях.

Каждая подпрограмма состоит из набора операторов, которые снабжены одним именем. По этому имени происходит **обращение** к данной подпрограмме.

С какими целями используют подпрограммы:

* сокращение записи программы (компактность);
* разделение работы между программистами, тем самым ускоряется ее завершение;
* упрощение внесений изменений в программу, так как исправление ошибки в одном блоке не вызывает корректировку других блоков;
* средство экономии памяти, так как память для хранения переменных, используемых в подпрограммах, выделяется только на время ее работы, и высвобождается после ее выполнения;
* использование подпрограмм позволяет реализовать один из самых прогрессивных методов программирования - структурное программирование.

**Вызовом** подпрограммы называется упоминание имени этой подпрограммы в теле основной программы с указанием списка фактических параметров, если они есть. Это приводит к активизации программной единицы, и выполнению входящих в неё операторов. После выполнения последнего из них управление возвращается обратно в основную программу, и выполняются операторы, стоящие непосредственно за оператором вызова процедуры.

**Практическая часть**

**Задание 1**

**Исходные данные**: Последовательность элементов задана общей формулой:

*a[i] = arctg (2 \* i + i / n) – sin (i + n)*, где *i* изменяется от 1 до *n*.

**Задание**

Составьте подпрограммы (по каждому пункту) для нахождения:

* Максимального элемента последовательности и его индекса;
* Количества элементов последовательности, превышающих по значению 1;
* Суммы элементов этой последовательности с четными индексами;
* *n* = 10.

**Программная реализация**

**program** zad1;

**var**

i, imax, n: integer;

A: **array of** real;

f: text;

max: real;

/////////////

**procedure** max\_element(A: **array of** real; n: integer; **var** imax: integer; **var** max: real);

**var**

i: integer;

**begin**

max:= A[0];

imax:= 0;

**for** i:= 1 **to** 9 **do**

**If** A[i] > max **then**

**begin**

max:= A[i];

imax:= i

**end**

**end**;

/////////////

**function** more\_than\_one(A: **array of** real): integer;

**var**

i: integer;

**begin**

result:= 0;

**for** i:= 0 **to** 9 **do**

**If** A[i] > 1 **then**

result:= result + 1;

**end**;

///////////////

**function** sum\_ch(A: **array of** real): real;

**var**

i: integer;

**begin**

result:= 0;

**for** i:= 0 **to** 9 **do**

**If** i **mod** 2 = 0 **then**

result:= result + A[i];

**end**;

//////////////

**begin**

n:= 10;

assign(f, 'task1output.txt');

rewrite(f);

SetLength(A, n);

////////////

**For** i:= 0 **to** 9 **do**

**begin**

A[i]:= arctan(2 \* i + i / n) - sin(i + n);

writeln(f, A[i]:4:2)

**end**;

max\_element(A, n, imax, max);

//////////////////

writeln(f, '1.');

writeln(f, 'максимальный член последовательности: ', max);

writeln(f, 'его порядковый номер равен: ', imax);

writeln(f, '2.');

writeln(f, 'количество членов последовательности, больших единицы: ', more\_than\_one(A));

writeln(f, '3.');

writeln(f, 'сумма членов последовательности с четными индексами: ', sum\_ch(A));

close(f);

**end**.

**Ответ:**

0.54

2.13

1.87

0.99

0.46

0.83

1.78

2.46

2.26

1.37

1.

максимальный член последовательности: 2.46427125392093

его порядковый номер равен: 7

2.

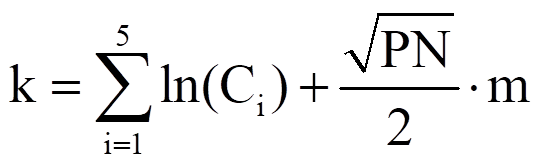
количество членов последовательности, больших единицы: 6

3.

сумма членов последовательности с четными индексами: 6.92117667350611

**Задание 2**

**Исходные данные**:



*m = 2,6e-4;*

*Ci = 3.42, 11.2, 0.4, 6.23, 15.64.*

**Задание**

Составьте процедуру для определения:

* Наибольшего значения (PN) произведений элементов каждой строки матрицы В(5,3);
* Элементы матрицы В(5, 3) заполните случайными числами из интервала [1; 100].
* Полученное значение PN используйте при вычислении: (исходные данные).

**В ответах приведите**

* Значения элементов матрицы В(5, 3);
* Значение PN;
* Значение k.

**Программная реализация**

**program** zad2;

**type**

arr = **array** [0..4, 0..2] **of** integer;

**var**

i, j: integer;

B: arr;

f: text;

k, m, sum: real;

**const**

C: **array** [0..4] **of** real = (3.42, 11.2, 0.4, 6.23, 15.64);

//////////////////////////////////

**function** PN\_val(B: arr): longint;

**var**

i, j: integer;

mult: integer;

C: **array** [0..4] **of** integer;

**begin**

mult:= 1;

**for** i:= 0 **to** 4 **do**

**begin**

**for** j:= 0 **to** 2 **do**

mult:= mult \* B[i, j];

C[i]:= mult;

mult:= 1;

**end**;

result:= C[0];

**for** i:= 1 **to** 4 **do**

**if** C[i] > result **then**

result:= C[i]

**end**;

//////////////////////////////////

**begin**

assign(f, 'task2output.txt');

rewrite(f);

writeln(f, 'Массив B:');

m:= 2.6e-4;

sum:= 0;

**for** i:= 0 **to** 4 **do**

**begin**

**for** j:= 0 **to** 2 **do**

**begin**

B[i, j]:= random(99) + 1;

write(f, B[i,j]:4)

**end**;

writeln(f)

**end**;

**for** i:= 0 **to** 4 **do**

sum:= sum + ln(C[i]);

k:= sum + sqrt(PN\_val(B)) / 2 \* m;

writeln(f, 'значение PN равно ', PN\_val(B));

writeln(f, 'значение k равно ', k);

close(f);

**end**.

**Ответ:**

Массив B:

48 15 65

5 40 15

31 16 38

25 36 60

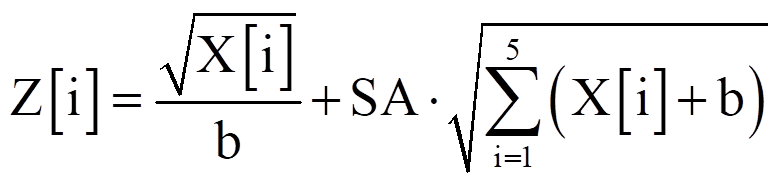
93 35 7

значение PN равно 54000

значение k равно 7.33868093551835

**Задание 3**

**Исходные данные**:



*b = 0.294;*

*X[i] = 10, 20, 30, 40, 50.*

**Задание**

Составьте подпрограмму–функцию для вычисления:

* Суммы (SA) значений среднего арифметического элементов каждого столбца матрицы А(3,4);
* Элементы матрицы А(3, 4) заполните случайными числами из интервала [1; 9];
* Значение SA используйте при вычислении: (исходные данные).

**В ответах приведите**

* Значения элементов матрицы A(5, 3);
* Значение SA;
* Значения элементов массива Z.

**Программная реализация**

**program** zad3;

**type**

arr = **array** [0..2, 0..3] **of** integer;

**var**

i, j: integer;

A: arr;

Z: **array** [0..4] **of** real;

f: text;

b, sum: real;

**const**

X: **array** [0..4] **of** integer = (10, 20, 30, 40, 50);

////////////////////////////////////

**function** SA\_val(A: arr): real;

**var**

i, j: integer;

sred, sum: integer;

**begin**

sred:= 0;

result:= 0;

**for** j:= 0 **to** 3 **do**

**begin**

**for** i:= 0 **to** 2 **do**

sred:= sred + A[i, j];

result:= result + sred / 3;

sred:= 0

**end**;

**end**;

////////////////////////////////////

**begin**

assign(f, 'task3output.txt');

rewrite(f);

writeln(f, 'Массив А:');

b:= 0.294;

sum:= 0;

**for** i:= 0 **to** 2 **do**

**begin**

**for** j:= 0 **to** 3 **do**

**begin**

A[i, j]:= random(8) + 1;

write(f, A[i,j]:2)

**end**;

writeln(f)

**end**;

**for** i:= 0 **to** 4 **do**

sum:= sum + X[i] + b;

**for** i:= 0 **to** 4 **do**

Z[i]:= sqrt(X[i]) / b + SA\_val(A) \* sqrt(sum);

writeln(f, 'значение SA равно ', SA\_val(A));

writeln(f, 'Массив Z:');

**for** i:= 0 **to** 4 **do**

write(f, Z[i]:8:2);

close(f);

**end**.

**Ответ:**

Массив А:

6 1 7 1

2 1 3 1

5 6 4 4

значение SA равно 13.6666666666667

Массив Z:

178.96 183.41 186.83 189.71 192.25

**Выводы**

В ходе работы успешно были составлены программы с помощью массивов как формальных параметров подпрограмм.