

Лабораторная работа № 6

«Детерминированные циклические вычислительные процессы с управлением по индексу»

Цель: разработать и научиться использовать алгоритмы, основанные на детерминированных циклических вычислительных процессах, управление которыми осуществляется индексом.

Оборудование: ПК, среда разработки «PascalABC»

Задание 1

Постановка задачи:

Исследовать характер изменения фазового угла φ и реактивного сопротивления колебательного контура Z на различных частотах f задана массивом значений

$$\varphi_i = \arctg\left(\frac{X_L}{R} - \frac{X_L^2}{RX_C} - \frac{R}{X_C}\right) \quad Z_i = X_C \sqrt{X_L^2 + R^2} / \sqrt{R^2 + (X_L - X_C)^2}, \quad \text{где}$$

$$X_C = \frac{1}{\omega_i C} = \frac{1}{2\pi f_i C} \quad X_L = 2\pi f_i L$$

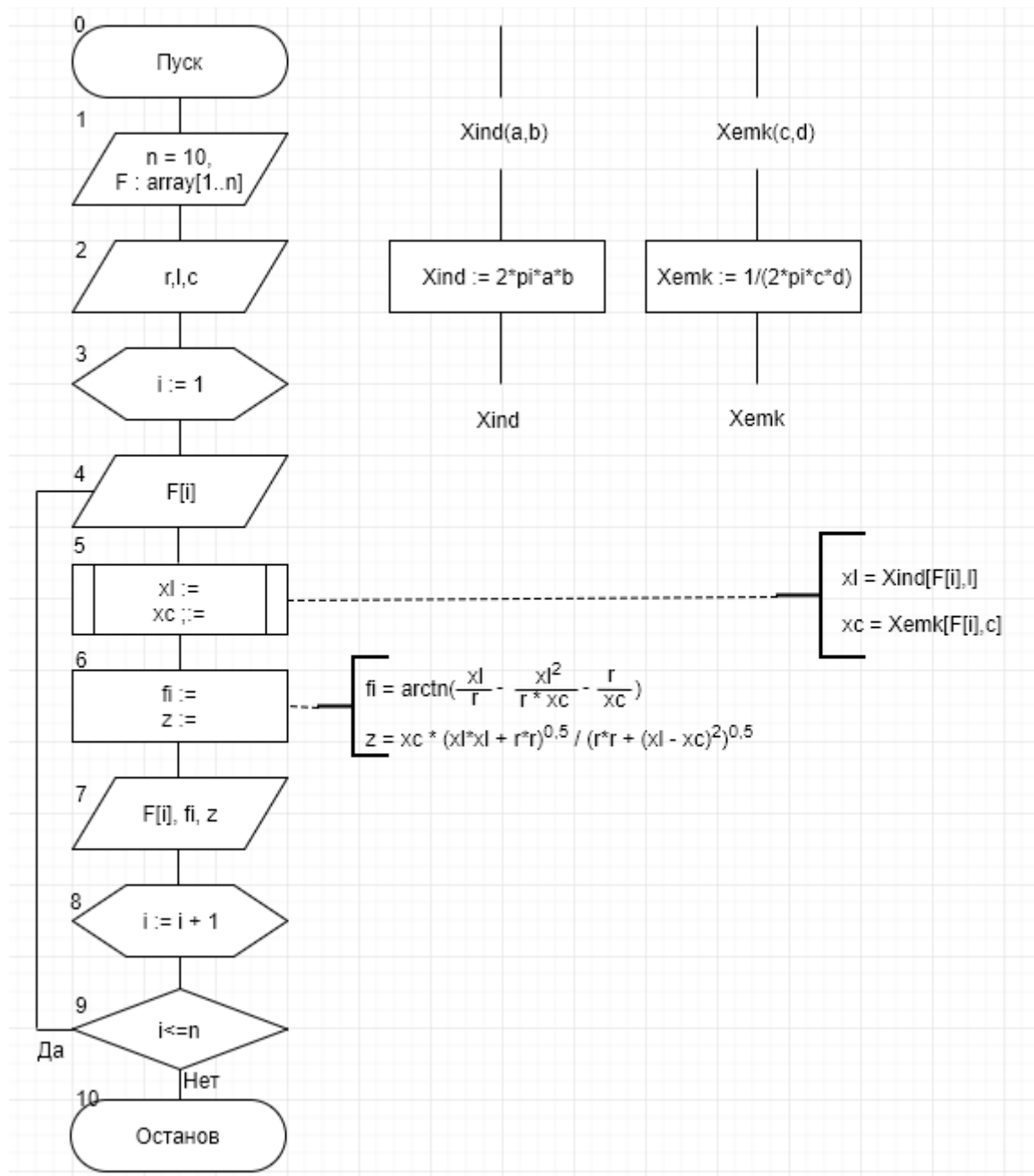
Значения R , C , L , F задать самостоятельно (значения должны быть реальными, посмотреть справочники и учебник по физике).

Математическая модель:

$$\varphi_i = \arctg \left(\frac{X_L}{R} - \frac{X_L^2}{R X_C} - \frac{R}{X_C} \right) \quad Z_i = X_C \sqrt{X_L^2 + R^2} / \sqrt{R^2 + (X_L - X_C)^2}, \quad \text{где}$$

$$X_C = \frac{1}{\omega_i C} = \frac{1}{2 \pi f_i C} \quad X_L = 2 \pi f_i L$$

Блок-схема:



Список идентификаторов:

Название	Тип	Функция
n	integer	Хранения значения кол-ва элементов
F	real	Массив данных
i	integer	Управление циклом
r	real	Хранение значения сопротивления
c	real	Хранение значения емкости
l	real	Хранение значения индуктивности
xl	real	Хранение значения выражения X_C
xc	real	Хранение значения выражения X_L
a	real	Передача функции Xind значения F[i]
b	real	Передача функции Xind значения l
c	real	Передача функции Xemk значения F[i]
d	real	Передача функции Xemk значения C
fi	real	Хранение значения фазового угла
z	real	Хранение значения реактивного сопротивления

Код программы:

```

] Program Zadaniel;
  const
    n = 10;
  Var
    F: Array [1..n] of real ;
    i : integer;
    r,l,xl,xc, fi,z,c : real;
] Function Xind(a,b : real) : real;
  Begin
    Xind := 2 * Pi * a * b;
  End;
-
] Function Xemk(c,d :real) : real;
  Begin
    Xemk := 1 / (2 * Pi * c * d);
  End;
-
] begin
  writeln('Введите сопротивление резистора:');
  readln(r);
  writeln('Введите индуктивность катушки:');
  readln(l);
  writeln('Введите емкость конденсатора:');
  readln(c);
  For i := 1 to n do
    Begin
      writeln('Введите значение частоты' );
      readln(F[i]);
      xl := Xind(F[i],l);
      xc := Xemk(F[i],c);
      fi := arctan((xl/r) - (xl*xl/(r*xc)) - (r/xc));
      z := xc * sqrt(xl*xl + r*r) / sqrt(r*r + (xl - xc)*(xl - xc));
      writeln('Значение частоты :', F[i]);
      writeln('Значение реактивного сопротивления :', z);
      writeln('Значение фазового угла :', fi);
    End;
  End.
- End.

```

Результаты вычислений:

Окно вывода

```
Введите сопротивление резистора:
2000
Введите индуктивность катушки:
0.000015
Введите емкость конденсатора:
0.000050
Введите значение частоты
15000
Значение частоты :15000
Значение реактивного сопротивления :0.212206605510181
Значение фазового угла :-1.57069022354496
Введите значение частоты
250
Значение частоты :250
Значение реактивного сопротивления :12.7321383977903
Значение фазового угла :-1.56443021459666
Введите значение частоты
450
Значение частоты :450
Значение реактивного сопротивления :7.07350931644013
Значение фазового угла :-1.5672595647649
Введите значение частоты
600000
Значение частоты :600000
Значение реактивного сопротивления :0.00530516516728077
Значение фазового угла :-1.57079367633119
Введите значение частоты
50
Значение частоты :50
Значение реактивного сопротивления :63.6297549572788
Значение фазового угла :-1.53897607977246
Введите значение частоты
```

Анализ результатов вычисления: Получение ответа на эту задачу стало возможным только благодаря использованию детерминированного циклического вычислительного процесса, управление которым осуществляет переменная «i» типа «integer». Прежде чем приступить к каким-либо вычислениям программа считывает с клавиатуры 3 показания: сопротивление, индуктивность и емкость, которые заносятся соответственно в переменные «r», «l», «c» типа «real». Первое действие, совершаемое программой в цикле – считывание значения частоты с клавиатуры. Это значение заносится в массив «F» типа «real». С целью сделать более удобный и читабельный код, я ввожу две функции, которые буду считать значения X_C и X_L . Функции « X_C » передаются значения частоты и емкости в переменные «a» и «b» типа «real». А функции « X_L » передаются значения частоты и индуктивности в переменные «c» и «d» типа «real». После совершения вычислений функции передают итоговые значения в переменные «xc» и «xl» типа «real». Следующим этапом происходит вычисление итоговых значений,

которые заносятся в переменные «z» и «fi» типа «». Заключительным действием цикла является вывод следующих данных: «F[i]», «z» и «fi»

Задание 2

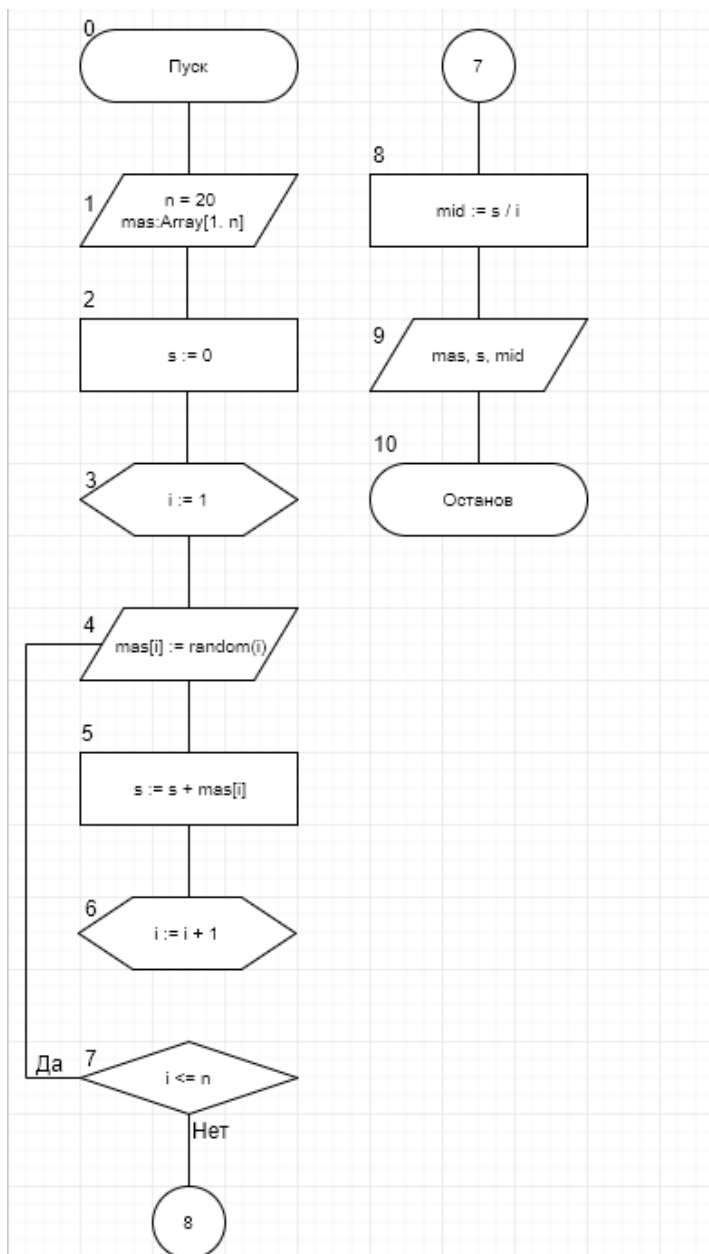
Постановка задачи: Одномерный массив задан случайным образом. Организовать вывод массива. Найти сумму его элементов, вычислить среднее арифметическое его элементов.

Математическая модель:

$$\sum mas = mas_1 + mas_2 + \dots + mas_n$$

$$\text{Ср. арифм} = \frac{\sum mas}{n}$$

Блок-схема:



Список идентификаторов:

Название	Тип	Функция
n	integer	Хранения значения кол-ва элементов
mas	integer	Массив данных
s	integer	Хранение значения суммы элементов
i	integer	Управление циклом
mid	real	Хранение значения среднего арифметического

Код программы:

```
Program zadanie2;
Const
  n = 20;
Var
  mid : real;
  i, s : integer;
  mas : Array[1..n] of integer;
begin
  s := 0;
  Randomize();
  For i := 1 to n do
  begin
    mas[i] := random(i);
    s := s + mas[i];
  end;
  mid := s / i;
  writeln('Массив : ', mas);
  writeln('Сумма элементов: ', s);
  writeln('Среднее арифметическое элементов: ', mid);
end.
```

Результаты вычислений:

```
Окно вывода
Массив : [0,0,2,3,0,3,2,2,5,6,4,7,8,13,11,13,10,14,16,9]
Сумма элементов: 128
Среднее арифметическое элементов: 6.4
```

Анализ результатов вычисления: Для решение этой задачи был применен циклический вычислительный процесс управление которым осуществлялось при помощи изменения индекса массива(переменная «i» типа «integer»). Перед началом массива происходит обнуление переменной «s» типа «integer», которая считает сумму элементов массива. В самом же цикле, происходят присваивания значений элемента массива, сразу же после этого

происходит и изменение суммы элементов по рекуррентной формуле. После выполнения этих действий индекс увеличивается на единицу по сравнению с предыдущим, потому была использована конструкция «For». После завершения цикла, происходит обычное линейное вычисление – находится среднее арифметическое элементов и записывается в переменную «mid» типа «real». Заключительным действием программа выводит сам массив, сумму его элементов и их среднее арифметическое.

Задание 3

Постановка задачи: Одномерный массив вводится пользователем с клавиатуры. Переставить элементы массива, стоящие на четных и нечетных местах. задачу решить без проверки на четность индексов массива.

Математическая модель:

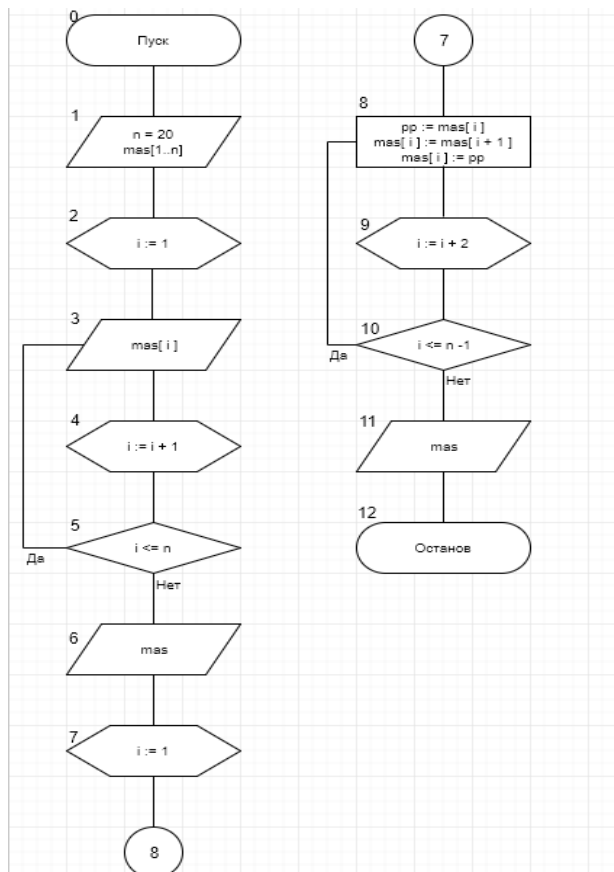
$$pp = mas_i$$

$$mas_i = mas_{i+1}$$

$$mas_{i+1} = pp$$

$$i = i + 2$$

Блок-схема:



Список идентификаторов:

Название	Тип	Функция
n	integer	Хранения значения кол-ва элементов
mas	integer	Массив данных
i	integer	Управление циклом
pp	integer	Хранение значения i-того элемента массива

Код программы:

```
Program Zadanie3;
Const
  n = 10;
Var
  i,pp :integer;
  mas : array [1..n] of integer;
begin
  For i := 1 to n do
  begin
    writeln('Введите ', i, '-тый элемент массива');
    readln(mas[i]);
  end;
  writeln('Введенный массив: ',mas);
  i := 1;
  while i <= n-1 do
  begin
    pp := mas[i];
    mas[i] := mas[i + 1];
    mas[i + 1] := pp;
    i := i + 2;
  end;
  writeln('Массив с перестановкой: ',mas);
end.
```

Результаты вычислений:

Окно вывода

```
Введите 1-тый элемент массива
1
Введите 2-тый элемент массива
2
Введите 3-тый элемент массива
3
Введите 4-тый элемент массива
4
Введите 5-тый элемент массива
5
Введите 6-тый элемент массива
6
Введите 7-тый элемент массива
7
Введите 8-тый элемент массива
8
Введите 9-тый элемент массива
9
Введите 10-тый элемент массива
10
Введенный массив: [1,2,3,4,5,6,7,8,9,10]
Массив с перестановкой: [2,1,4,3,6,5,8,7,10,9]
```

Анализ результатов вычисления: Для решения данной задачи были выполнены вычисления, основанные на детерминированных циклических процессах. Первый цикл, управление которым осуществляет переменная «i» типа «integer», изменяющаяся по рекуррентной зависимости ($i = i + 1$), предназначен для ввода массива с клавиатуры. После завершения первого цикла, программа выполнит вывод полученного массива на экран, в целях большего удобства. Следующим шагом происходит пере присваивание значения переменной «i». Управление вторым циклом также осуществляется так-же при помощи переменной «i», но ее изменение описывается уже другой рекуррентной зависимостью (). В самом цикле вычисления происходят в несколько этапов. Первый — это присвоение переменной «pr» типа «integer» значения i-того элемента массива. Второй – это присвоение i-тому элемента массива значение элемента с индексом i+1. Третий – это присвоение i+1- ому элементу массива значение i-того элемента. После выполнения цикла, мы выводим уже измененный массив.

Задание 4

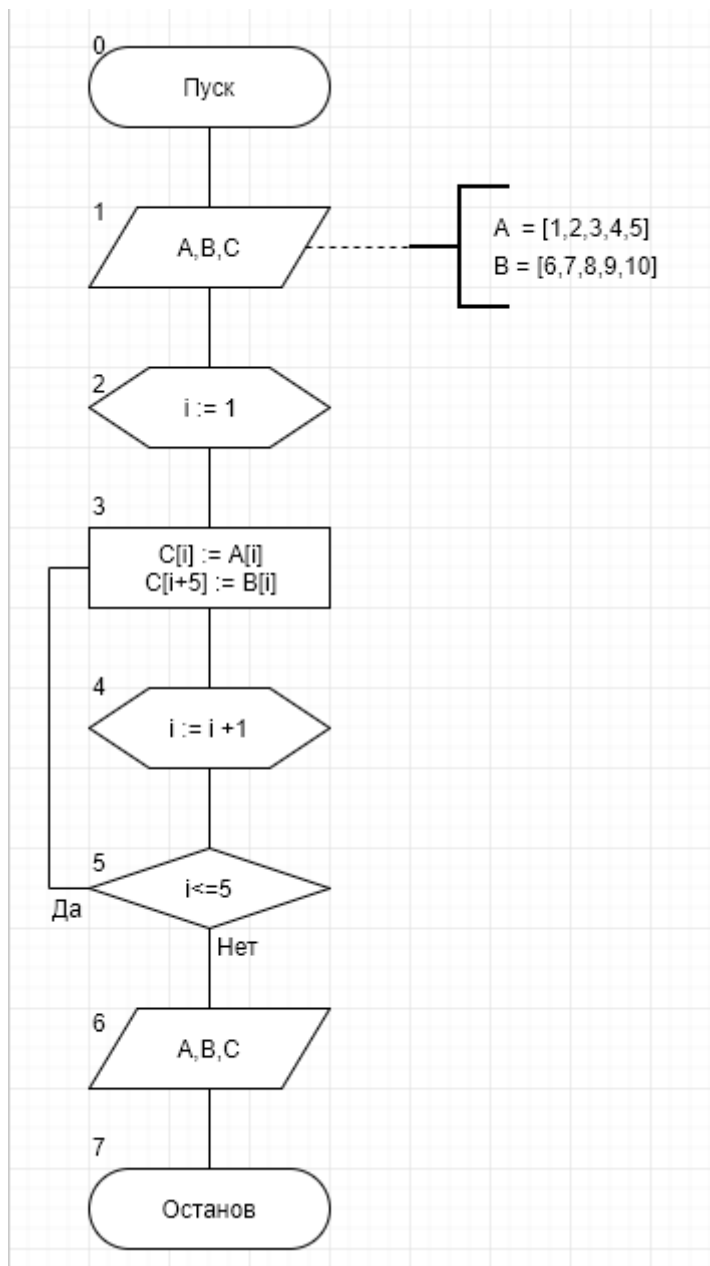
Постановка задачи: Заданы массивы A(5) и B(5). Получить массив C(10), расположив вначале его элементы массива A, а затем – элементы массива B. Для формирования массива C использовать один цикл.

Математическая модель:

$$C_i = A_i$$

$$C_{i+5} = B_i$$

Блок-схема:



Список идентификаторов:

Название	Тип	Функция
A	integer	Массив A
B	integer	Массив B
C	integer	Массив C
i	integer	Управление циклом

Код программы:

```
Program Zadanie4;
const
  A : array [1..5] of integer = (1,2,3,4,5);
  B : array [1..5] of integer = (6,7,8,9,10);
Var
  C : array [1..10] of integer;
  i :integer;
begin
  For i := 1 to 5 do
  begin
    C[i] := A[i];
    C[i + 5] := B[i];
  end;
  Writeln('Массив A: ', A);
  Writeln('Массив B: ', B);
  Writeln('Массив C: ', C);
end.
```

Результаты вычислений:

Окно вывода

```
Массив A: [1,2,3,4,5]
Массив B: [6,7,8,9,10]
Массив C: [1,2,3,4,5,6,7,8,9,10]
```

Анализ результатов вычисления: Вычислительный процесс, используемый для решения данной задачи, относится к категории детерминированных вычислительных управление которыми осуществляется по индексу.

Индексом массив служит переменная «i» типа «integer», изменяющаяся по рекуррентной формуле ($i = i + 1$). Все вычисления данной задачи происходят в цикле, тк массивы «A» и «B» являются инициализированными, в связи с тем, что в условии не прописано как именно мы их заполняем. Вычисления происходят следующим образом: Массив «C» мысленно разделяется на 2 части и начинает заполняться с i-того элемента и с $i + 5$ -того элемента. Первым 5 элементам массивы присваивают значения элементов массива «A» а следующим 5, значения элементов массива «B». Использование именно такого способа позволяет использовать всего 1 цикл, что существенно снижает затраты времени.

Вывод: Детерминированные вычислительные процессы с управлением по индексу, являются очень важным элементом программирования, тк благодаря им можно решать огромное кол-во задач, связанных с массивами. Огромный спектр задач можно решить используя массив данных, и как следствие без знания такой вещи как ДЦВП с управлением по индексу, их совершенно невозможно будет решить.