

Спецификация к Лабораторным работам №2-4 (Вариант 22)
Абстракция А0

1. Постановка задачи (ПЗ).

Задание: Написать программу обработки одномерного массива в соответствии с условием.

Условие: При заданных X_1, X_2, \dots, X_n сформировать массив элементов Y_1, Y_2, \dots, Y_n по правилу:

$$Y_i = \begin{cases} 1 - \sin X_i & | X_i > 0 \\ 1 - \cos X_i & | X_i \leq 0 \end{cases} \quad (1). \text{ При этом подсчитать число неотрицательных } X_i.$$

2. Уточненная постановка задачи.

Дан *одномерный вещественный* массив X из n ($0 < n \leq 20$) элементов.

Найти:

Y – *одномерный вещественный* массив, элементы которого подсчитаны по правилу (1);

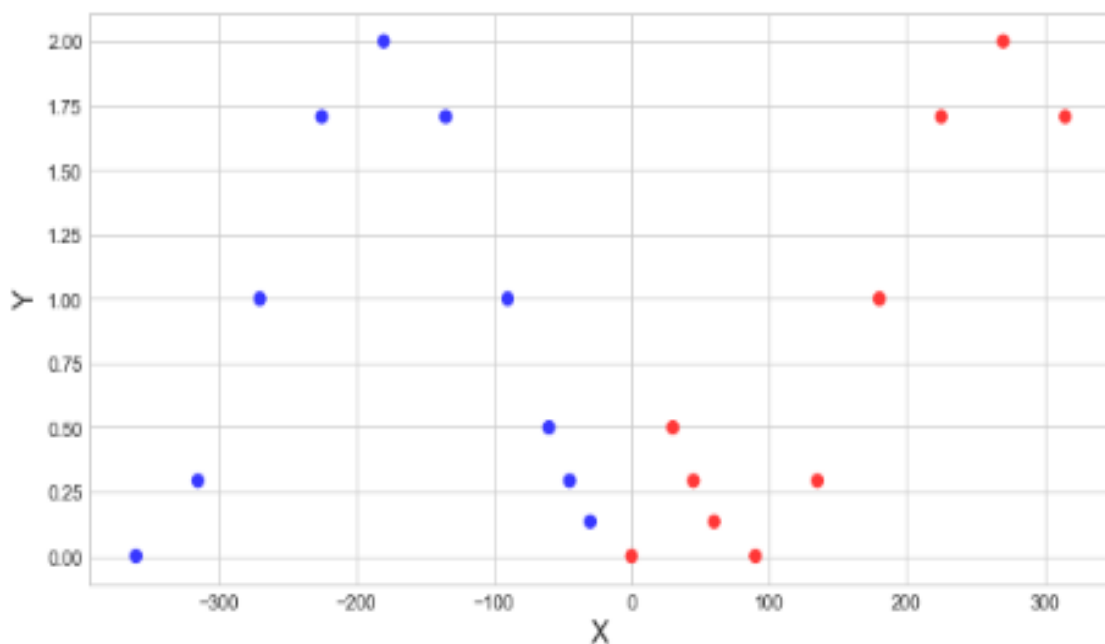
c – количество *неотрицательных* элементов массива X .

3. Пример с иллюстрацией.

```

                                Лаб. 2
Количество элементов массива X (n): 20
Массив X:
-360
-315
-270
-225
-180
-135
-90
-60
-45
-30
0
30
45
60
90
135
180
225
270
315
=====
Массив Y:
0.0
0.292893
1.0
1.707107
2.0
1.707107
1.0
0.5
0.292893
0.133975
0.0
0.5
0.292893
0.133975
0.0
0.292893
1.0
1.707107
2.0
1.707107

Количество неотрицательных элементов массива X (c): 10
```



4. Таблица данных

Класс	Имя	Описание (смысл, диапазон, точность)	Тип	Структура	Формат в/в
Входные данные	<i>n</i>	число элементов массива, $0 < n \leq 20$	цел	простая переменная	XX (:2)
	<i>X</i>	вводимые с клавиатуры числа	вещ	одномерный массив (20)	+XX.XX+ (:5:6)

Выходные данные	<i>Y</i>	числа, подсчитанные по правилу (1)	вещ	одномерный массив (20)	X.XX+ (:1:6)
	<i>c</i>	число неотрицательных элементов массива, $0 < c \leq 20$	цел	простая переменная	XX (:2)
Промежу- точные данные*	<i>i</i>	индекс текущего элемента, $0 \leq i \leq 21$ <i>(в Delphi выход из цикла с 1 до 20 с шагом +1 происходит при значении 21, из цикла с 20 до 1 – при 0)</i>	цел	простая переменная	---

5. Входная форма

обр 1.1 Количество элементов массива X (n) :
 обр 1.2 <n>
 обр 2.1 Массив X:
 обр 2.2 <x[1]>
 <x[2]>
 ...
 <x[n]>

6. Выходная форма

обр 3 Лаб. 2

обр 4 Количество элементов массива X (n): <n>

обр 5.1 Массив X:

обр 5.2 <X[1]>

<X[2]>

...

<X[n]>

обр 6.1 Массив Y:

обр 6.2 <Y[1]>

<Y[2]>

...

<Y[n]>

обр 7 Количество неотрицательных элементов массива X (c): <c>

7. Аномалии

№	Описание	Условие возникновения**	Реакция на аномалию
1	<i>n</i> меньше минимально допустимого значения	$n < 1$	Сообщение: «Некорректное <i>n</i> : $n < 1$ » (обр.А1) Действие: Завершение работы программы
2	<i>n</i> больше максимально допустимого значения	$n > 20$	Сообщение: «Некорректное <i>n</i> : $n > 20$ » (обр.А1) Действие: Завершение работы программы

8. Функциональные тесты

Исходные данные		Результаты			Тест №
n	X	Y	c	Сообщения	
20	[-360, -315, -270, -225, -180, -135, -90, -60, -45, -30, 0, 30, 45, 60, 90, 135, 180, 225, 270, 315]	[0.0, 0.29289322, 1.0, 1.70710678, 2.0, 1.70710678, 1.0, 0.5, 0.29289322, 0.133974, 0.0, 0.5, 0.29289322, 0.1339746, 0.0, 0.29289322, 1.0, 1.70710678, 2.0, 1.70710678]	10	-	1
1	[90]	[0.0]	1	-	2
4	[0, -10, -100, -1000]	[0.0, 0.015192, 1.173648, 0.826352]	1	-	3
22	-	-	-	Некорректное <i>n</i> : $n > 20$	4
0	-	-	-	Некорректное <i>n</i> : $n < 1$	5

№ теста	Входные данные	Ожидаемый результат	Смысл теста
1	n = 20 X = [-360, -315, -270, -225, -180, -135, -90, -60, -45, -30, 0, 30, 45, 60, 90, 135, 180, 225, 270, 315]	Y = [0.0, 0.29289322, 1.0, 1.70710678, 2.0, 1.70710678, 1.0, 0.5, 0.29289322, 0.133974, 0.0, 0.5, 0.29289322, 0.1339746, 0.0, 0.29289322, 1.0, 1.70710678, 2.0, 1.70710678] c = 10	Подтвердить правильность расчетов выходных данных
2	n = 1 X = [90]	Y = [0.0] c = 1	Протестировать простейший случай
3	n = 1 X = [0,-10,-100,-1000]	Y = [0.0, 0.0151926, 1.173648, 0.826352] c = 1	Проверить правильность выполнения программы при разделении переменных на положительные и неположительные, на отрицательные и неотрицательные.
4	n = 22	“Некорректное n: n>20”	Проверить корректность определения аномалии (обр.А1)
5	n = 0	“Некорректное n: n<1”	Проверить корректность определения аномалии (обр.А1)

Результаты		№ теста
Y	Максимальная вычислительная нагрузка – 20 элементов нового массива. Соответствует ожидаемым числовым значениям.	1
	0	2
	Максимальная вычислительная нагрузка – 4 элемента нового массива. Соответствует ожидаемым числовым значениям.	3
	Элементы нового массива не вычисляются.	4
	Элементы нового массива не вычисляются.	5
c	Максимальная вычислительная нагрузка – подсчет количества неотрицательных элементов массива входных данных. Соответствует ожидаемым числовым значениям.	1
	1	2
	Правильно определены положительные и неотрицательные элементы массива входных данных	3
	Обработка входного массива не производится.	4
	Обработка входного массива не производится.	5
Сообщения	Вывод на экран входных и выходных данных.	1
	Вывод на экран входных и выходных данных.	2
	Вывод на экран входных и выходных данных.	3
	Аномалия определена верно.	4
	Аномалия определена верно.	5

9. Метод

Отделим ввод-вывод от обработки данных и разделим задачу на три подзадачи:

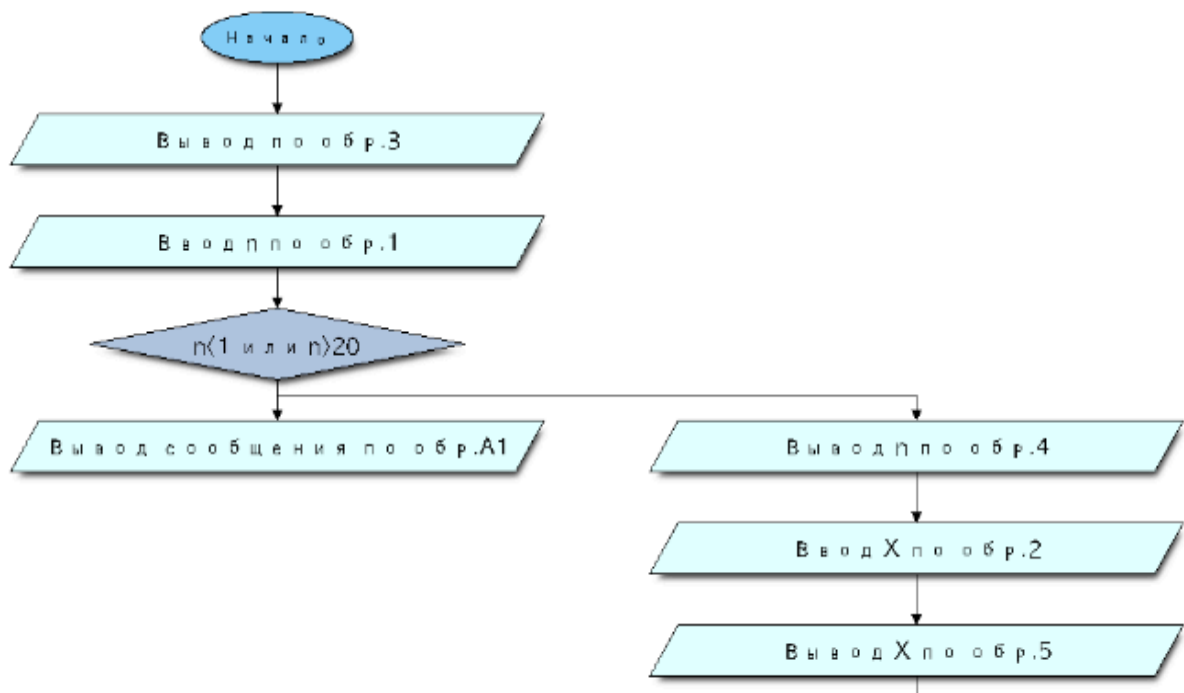
1. Подзадача А0.1. Ввести исходные данные (обр.1,2) и вывести их на экран (обр.3-5) для визуального подтверждения.
2. Подзадача А0.2. Решение поставленной задачи: на основе введенных исходных данных (n , X) найти новый массив Y и количество неотрицательных элементов исходного массива c .
3. Подзадача А0.3. Ввести вывести полученные результаты (Y, c) на экран (обр.6,7)

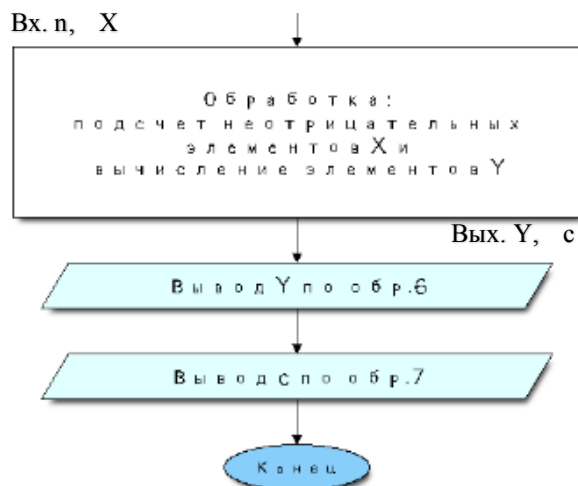
Ввод-вывод тривиален и представляет собой чисто техническую задачу.

Для осуществления процесса ввода-вывода массивов потребуется промежуточная переменная i – индекс текущего элемента массива, которую следует добавить в таблицу данных.

Алгоритм необходим только для обработка входных данных и расчета элементов нового массива.

10. Алгоритм





11. Программа на *Delphi*.

Диалоговый вариант {ввод с клавиатуры, вывод на экран}

```

program Lab2; //сохранить как lab2.dpr (DPR – Delphi PProject)
{$APPTYPE CONSOLE}
Uses Windows; // для русификации
const
  Nmax=20; {верх. граница индексов массива}
var
  n, i, c: integer;
  X, Y: array [1..Nmax] of real;

begin
  setConsoleCP(1251); // для ввода русификация и
  setConsoleOutputCP(1251); // для вывода

  {A0.1 - ввод-вывод входных данных}
  writeln('Лаб.2' :40); {вывод заголовка с переходом на следующую строку по обр.3}

  writeln('Количество элементов n:');
  readln(n); {ввод-вывод цел. переменной - количество элементов по обр.1,4}

  if(n<1) or (n>20) then
    if (n<1) then
      write('Некорректное n: n<1')
    else
      write('Некорректное n: n>20')
  else

    begin
      {ввод-вывод исходного массива X по обр.2,5}
      writeln('Массив X: '); {вывод пояснения с переходом на следующую строку}
      for i:=1 to n do
        begin
          readln(X[i]); {ввод элемента и переход на следующую строку}
          writeln(X[i]:5:6);{вывод элемента и переход на следующую строку}
        end;
      for i:=1 to 80 do
        write('='); {отделим визуально чертой и строкой введенные и выводимые значения}
      writeln;

      {A0.2 - обработка, A0.3 - вывод результатов}
      c := 10; {заглушка для теста 1}
    end
  end;

```

```

Y[1] := 0.0; Y[2] := 0.292893; Y[3] := 1.0; Y[4] := 1.707107; {заглушка для теста 1}
Y[5] := 2.0; Y[6] := 1.707107; Y[7] := 1.0; Y[8] := 0.5; {заглушка для теста 1}
Y[9] := 0.292893; Y[10] := 0.133975; Y[11] := 0.0; Y[12] := 0.5; {заглушка для теста 1}
Y[13] := 0.292893; Y[14] := 0.133975; Y[15] := 0.0; Y[16] := 0.292893; {заглушка для теста 1}
Y[17] := 1.0; Y[18] := 1.707107; Y[19] := 2.0; Y[20] := 1.707107; {заглушка для теста 1}

//      с := 1; Y[1] := 0.0; {заглушка для теста 2}
//      с := 1; Y[1] := 0.0; Y[2] := 0.015192; Y[3] := 1.173648; Y[4] := 0.826352; {заглушка для теста 3}

{вывод элементов массива Y по обр.6}
writeln('Массив Y: ');
for i:=1 to n do
begin
  writeln(Y[i]:1:6);
end;

writeln('Количество неотрицательных элементов X:', с){вывод переменной по обр.7};

write('Press Enter...'); readln; {задерживаем экран до нажатия ENTER}
end;
end.

```

Раскрытие абстракции A0.2-A0.3

*Поскольку условие всей задачи **совпадает** (за исключением необходимости ввода) с условием выделенной подзадачи, вместо полной спецификации с пунктами 1-11 выполнено просто дополнение к пунктам 9-11.*

9. Метод

Пусть

i – номер текущей точки,

$X[i]$ – соответствующий элемент исходного массива X ,

$Y[i]$ – полученный по правилу (1) элемент массива Y .

c - количество неотрицательных элементов массива X .

Присвоим переменной c начальное значение: $c = 0$.

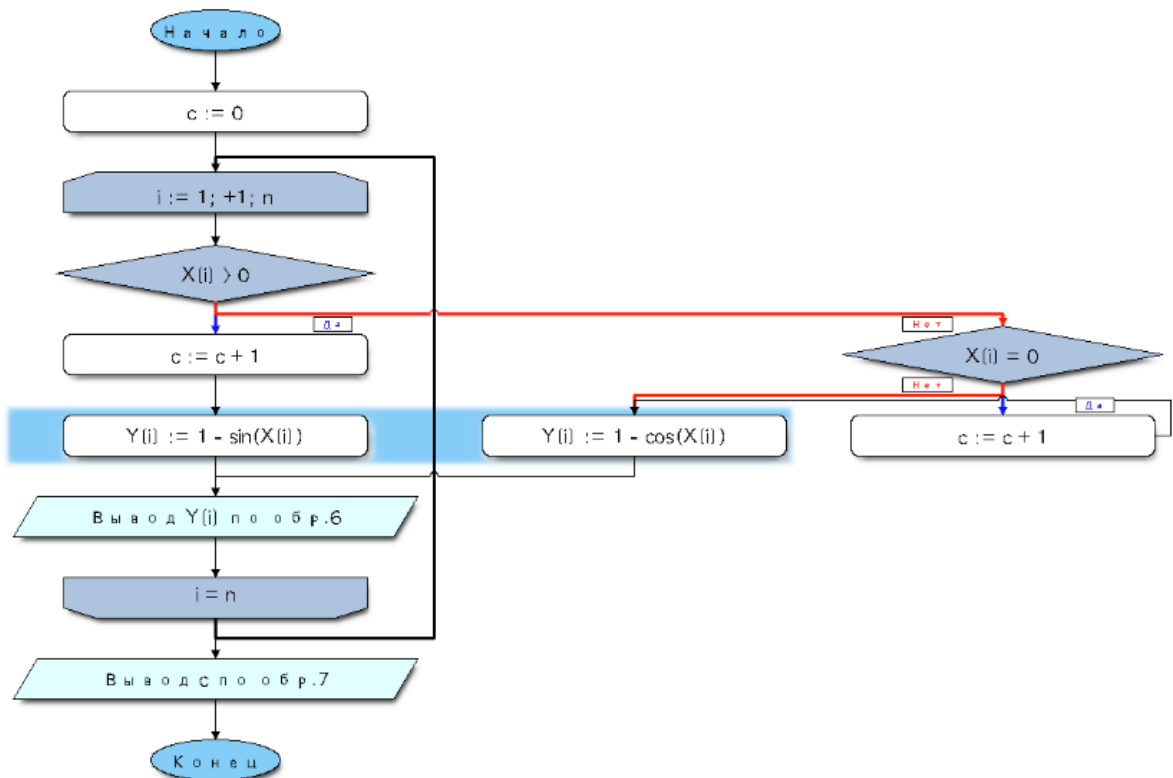
При $X[i] > 0$ получим $Y[i] = 1 - \sin(X[i])$.

Переменная c при этом увеличивается на 1.

При $X[i] \leq 0$ получим $Y[i] = 1 - \cos(X[i])$.

Переменная c при этом увеличивается на 1 только при $X[i] = 0$.

10. Алгоритм



11. Программный код

{A0.2 - обработка, A0.3 - вывод результатов}
 c := 0;

{вывод элементов массива Y по обр.6}
 writeln('Массив Y: '); {вычисление и вывод элементов массива}

```

for i:=1 to n do
begin
  if (X[i] > 0) then
  begin
    Y[i] := 1 - sin(X[i]);
    c := c + 1;
  end
  else
  begin
    if (X[i] = 0) then c := c + 1;
    Y[i] := 1 - cos(X[i]);
  end;
  writeln(Y[i]:1:6);
end;

```

writeln('Количество неотрицательных элементов X:', c){вывод переменной по обр.7};

Получившиеся программы

Лабораторная работа 3


```

program Lab3; //сохранить как lab3.dpr (DPR – Delphi PRoject)
{$APPTYPE CONSOLE}
Uses Windows; // для русификации (сделать комментарием в онлайн-компиляторе!)
const
    Nmax=20; {верх. граница индексов массива}
var
    n, i, c: integer;
    X, Y: array [1..Nmax] of real;

begin
    setConsoleCP(1251); // для ввода русификация и (сделать строку комментарием в
онлайн-компиляторе!)
    setConsoleOutputCP(1251); // для вывода (сделать строку комментарием в онлайн-
компиляторе!)

    {A0.1 - ввод-вывод входных данных}
    writeln('Лаб.2' :40); {вывод заголовка с переходом на следующую строку по обр.3}

    writeln('Количество элементов n:'); readln(n); {ввод-вывод цел. переменной -
количество элементов по обр.1,4}

    if(n<1) or (n>20) then
        if (n<1) then
            write('Некорректное n: n<1')
        else
            write('Некорректное n: n>20')
        else

begin
    {ввод-вывод исходного массива X по обр.2,5}
    writeln('Массив X: '); {вывод пояснения с переходом на следующую строку}
    for i:=1 to n do
        begin
            readln(X[i]); {ввод элемента и переход на следующую строку}
            writeln(X[i]:5:6); {вывод элемента и переход на следующую строку}
        end;
    for i:=1 to 80 do
        write('='); {отделим визуально чертой и строкой введенные и выводимые
значения}
    writeln;

    {A0.2 - обработка, A0.3 - вывод результатов}
    c := 0;
    {вывод элементов массива Y по обр.6}
    writeln('Массив Y: '); {вычисление и вывод элементов массива}
    for i:=1 to n do
        begin
            if (X[i] > 0) then
                begin
                    Y[i] := 1 - sin(X[i]);
                    c := c + 1;

```

```

        end
      else
      begin
        if (X[i] = 0) then c := c + 1;
        Y[i] := 1 - cos(X[i]);
      end;
      writeln(Y[i]:1:6);
    end;

    writeln('Количество неотрицательных элементов X:', c){вывод переменной по
обр.7};

    write('Press Enter...'); readln; {задерживаем экран до нажатия ENTER}
  end;
end.

```

Лабораторная работа 4

```

program Lab4; //сохранить как lab4.dpr (DPR – Delphi PProject)
{$APPTYPE CONSOLE}
Uses Windows; // для русификации (сделать комментарием в онлайн-компиляторе!)
const
  Nmax=20; {верх. граница индексов массива}
var
  n, i, c: integer;
  X, Y: array [1..Nmax] of real;
  fin, fout: TextFile; {описание файловых переменных, для работы с текстовыми
файлами} // lab.4

begin
  setConsoleCP(1251); // для ввода русификация и (сделать строку комментарием в
онлайн-компиляторе!)
  setConsoleOutputCP(1251); // для вывода (сделать строку комментарием в онлайн-
компиляторе!)

  if (ParamCount<2) then // lab.4
  begin
    writeln('CritError: Не заданы параметры программы');
    readln; exit;
  end;
  AssignFile(fin, ParamStr(1)); AssignFile(fout, ParamStr(2)); // lab.4
  Reset(fin); {открытие для чтения текстового файла, связанного с fin в предыдущей
строке} // lab.4
  Rewrite(fout); {создание и открытие для записи файла, связанного с переменной fout}
// lab.4

  {A0.1 - ввод-вывод входных данных}
  writeln(fout, 'Лаб.4' :40); {вывод в файл заголовка с переходом на следующую строку
по обр.3} // lab.4

  {ввод-вывод цел. переменной - количество элементов по обр.1,4}
  writeln(fout, 'Количество элементов n:'); readln(fin, n); // lab.4

```

```

if(n<1) or (n>20) then
  if (n<1) then
    write(fout,'Некорректное n: n<1') // lab.4
  else
    write(fout,'Некорректное n: n>20') // lab.4
  else
    begin
      {ввод-вывод исходного массива X по обр.2,5}
      writeln(fout,'Массив X: '); {вывод пояснения с переходом на следующую строку} //
lab.4
      for i:=1 to n do
        begin
          readln(fin, X[i]); {ввод элемента и переход на следующую строку} // lab.4
          writeln(fout, X[i]:5:6); {вывод элемента и переход на следующую строку} // lab.4
        end;
      {отделим визуально чертой и строкой введенные и выводимые значения}
      for i:=1 to 80 do
        write(fout, '='); // lab.4
      writeln(fout); // lab.4

      {A0.2 - обработка, A0.3 - вывод результатов}
      c := 0;
      {вывод элементов массива Y по обр.6}
      writeln(fout, 'Массив Y: '); // lab.4
      {вычисление и вывод элементов массива}
      for i:=1 to n do
        begin
          if (X[i] > 0) then
            begin
              Y[i] := 1 - sin(X[i]);
              c := c + 1;
            end
          else
            begin
              if (X[i] = 0) then c := c + 1;
              Y[i] := 1 - cos(X[i]);
            end;
          writeln(fout, Y[i]:1:6); // lab.4
        end;

      {вывод переменной по обр.7}
      writeln(fout, 'Количество неотрицательных элементов X:', c); // lab.4
    end;

    CloseFile(fin); CloseFile(fout); // lab.4
    write('Press Enter...'); readln; {задерживаем экран до нажатия ENTER}
  end.

```