Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра программного обеспечения информационных технологий

Дисциплина: Основы алгоритмизации и программирования (ОАиП)

ОТЧЕТ

по лабораторной работе №3

Тема работы:

Работа с одномерными массивами

Выполнил

студент: гр. 151004 Башлыков В.В.

Проверил: Фадеева Е. П.

Минск 2021

Содержание

[1 Постановка задачи 3](#_Toc91598752)

[2 Ход решения 4](#_Toc91598753)

[3 Текстовый алгоритм решения задачи 5](#_Toc91598754)

[4 Структура данных 6](#_Toc91598755)

[5 Схема алгоритма решения задачи по ГОСТ 19.701-90 7](#_Toc91598756)

[6 Результаты расчетов 10](#_Toc91598757)

[Приложение А 12](#_Toc91598758)

[Приложение Б 15](#_Toc91598759)

[Приложение В 17](#_Toc91598760)

[Приложение Г 20](#_Toc91598761)

[Приложение Д 23](#_Toc91598762)

# Постановка задачи

Ввести n и последовательность B[n] из целых чисел. Определить начальный и конечный номера элементов массива, составляющих последовательности, для которой значения элементов равны между собой.

Правила постановки:

1. По условию задачи двумерный массив Mas формируется из целых чисел;
2. Программа учитывает некорректный ввод пользователя:

* не числовые значения;
* переполнения значения Integer;

1. Программа предусматривает потенциальное переполнение переменной типа Integer при вычислениях.
2. Нельзя сортировать массив Mas.
3. Для решения задачи используется 4 постановки:

* ввод данных в массив выполняет пользователь с клавиатуры(№1);
* ввод данных в массив с использованием генератора чисел(№2);
* ввод данных в массив с использованием типизированной константы-массив(№3);
* ввод данных в массив с использованием типизированной переменной-массив(№4).

# Ход решения

1. Так как из данных элементов нужно выбрать те последовательности, где значения элементов равны, то необходимо сравнивать между собой значения соседних элементов в массиве.
2. Для получения верного результата необходимо сравнивать следующие элементы массива: Mas[I – 1], Mas[I], Mas[I + 1], где I номер текщего элемента.
3. Ели элемент массива Mas[I] равен элементу массива Mas[I – 1], значит в массиве есть последовательность как минимум из двух элементов.
4. Далее необходимо проверять каждый последующий элемент массива на выполнение условия Mas[I] = Mas[I – 1].
5. Далее необходимо определить элемент массива, на котором последовательность равных элементвов заканччивается.
6. Для нахождения последнего элемента массива нужно проверить следующее условие на истинность: Mas[I] <> Mas[I – 1] или Mas[I] <> Mas [I + 1].
7. Если Mas[I] <> Mas [I + 1], то последний элемент последовательности равных элементов будет равен Mas[I].
8. Если Mas[i] <> Mas[ I – 1], то последний элемент последовательности равных элементов будет равен Mas[I – 1].

# Текстовый алгоритм решения задачи

Таблица 1 – Алгоритм решения

|  |  |
| --- | --- |
| Номер  шага | Назначение шага |
|  | N := 20 |
|  | Ввод Mas[1..N] |
|  | I := 2 |
|  | Flag := False; |
|  | Stop := False; |
|  | Начало цикла А1. Проверка выполнения условия I <= Length(Mas) - 1. Если условие истинно, идти к шагу 7, иначе к шагу 25 |
|  | Начало цикла A2. Проверка выполнения условия (Flag = False) and (i <= Length(Mas)). Если условие истинно, идти к шагу 8, иначе идти к шагу 22 |
|  | CurrentValue := Mas[i]; |
|  | NextValue := Mas[i + 1]; |
|  | PreviouValue := Mas[i - 1]; |
|  | Проверка выполнения условия (CurrentValue = PreviouValue) and (Stop = False). Если условие истинно, идти к шагу 12, иначе идти к шагу 20 |
|  | First := i - 1; |
|  | Проверка выполнения условия ((CurrentValue <> NextValue) or (CurrentValue <> PreviouValue)) and (Stop = True). Если условие истинно, идти к шагу 14, иначе идти к шагу 20 |
|  | Проверка выполнения условия (CurrentValue <> NextValue). Если условие истинно, идти к шагу 15, иначе идти к шагу 16 |
|  | Last := i; |
|  | Проверка выполнения условия (CurrentValue <> NextValue). Если условие истинно, идти к шагу 17, иначе идти к шагу 18 |
|  | Last := i - 1; |
|  | Проверка выполнения условия (PreviouValue <> CurrentValue) or (NextValue <> CurrentValue). Если условие истинно, идти к шагу 19, иначе идти к шагу 20 |
|  | Flag := True; |
|  | I := I + 1; |
|  | Конец цикла А2.Идти к шагу 7 |
|  | Flag := False; |
|  | Stop := False; |
|  | Конец цикла А1.Идти к шагу 6. |
|  | Вывод First, Last, Mas[First..Last] |
| 25 | Останов. | |

# Структура данных

Таблица 2 – Данные

|  |  |  |
| --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение |
| N | Const | Размер массива Mas |
| Mas | Array [1..N] of  Integer | Массив принимающий введенные пользователем значения. |
| Flag | Boolean | Переменная для контроля использования двоичного выбора |
| Stop | Boolean | Переменная для контроля использования двоичного выбора |
| CurrentValue | Integer | Элемент массива Mas[I] |
| PreviouValue | Integer | Элемент массива Mas[I - 1] |
| NextValue | Integer | Элемент массива Mas[I + 1] |
| First | Integer | Переменная для хранения номера первого члена последовательности в массиве |
| Last | Integer | Переменная для хранения номера последнего члена последовательности в массиве |
| I | Integer | Счетчик цикла |

# Схема алгоритма решения задачи по ГОСТ 19.701-90

Рисунок 1 – Схема алгоритма решения задачи по ГОСТ 19.701-90

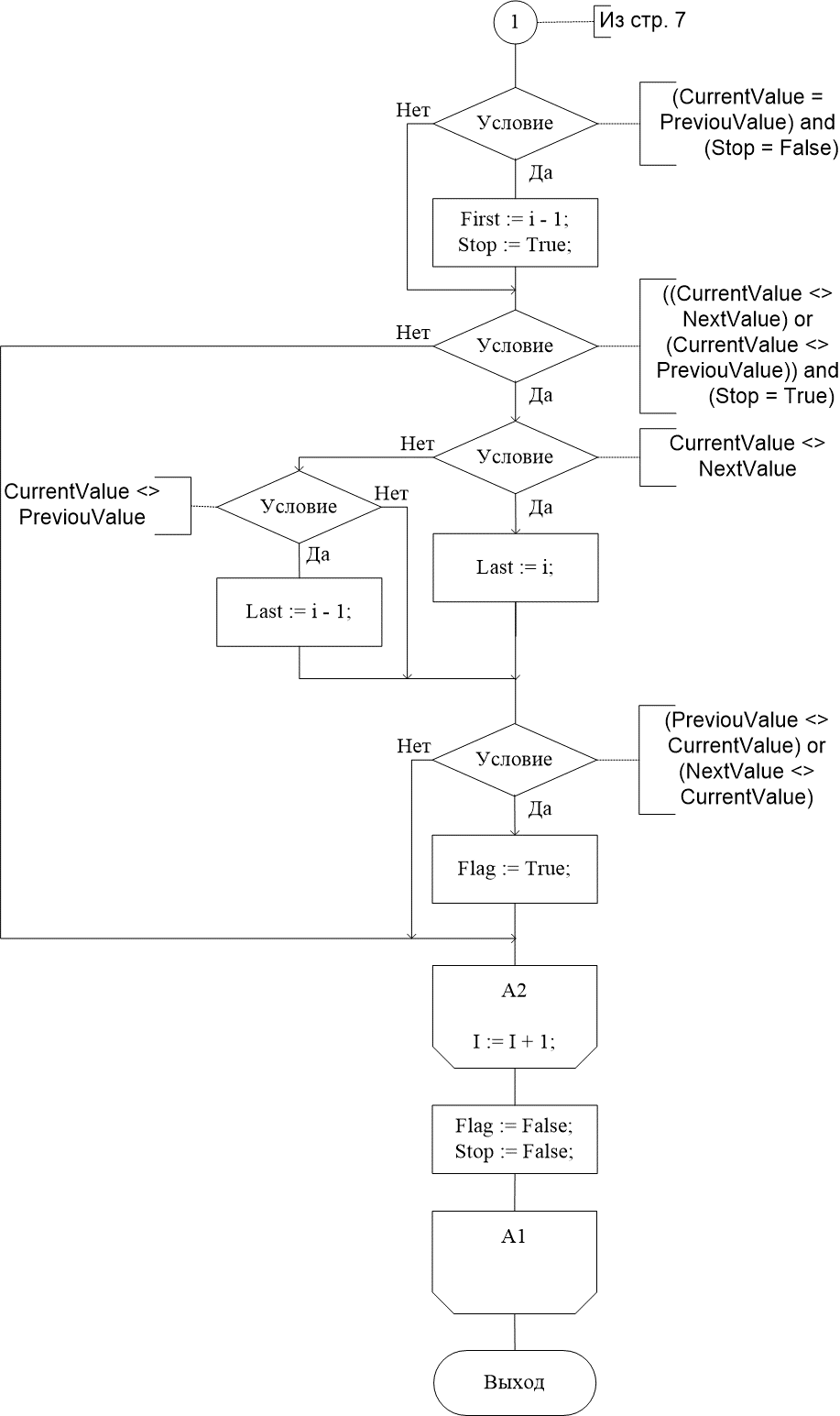


Рисунок 2 – Схема алгоритма решения задачи по ГОСТ 19.701-90

# Результаты расчетов

Вследствие выполнения программы на экран выводятся следующие результаты:

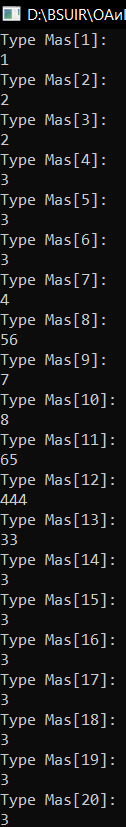


Рисунок 3 – Введённые данные

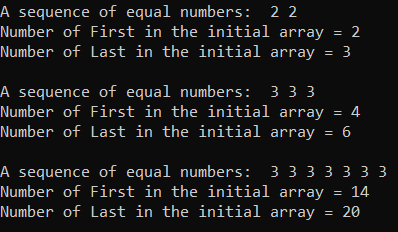


Рисунок 4 – Полученный результат

Приложение А

(обязательное)

Исходный код программы (по постоновке 1)

Массив заполняется путём ввода данных с клавиатуры

Program LBR\_3;

{Enter n and a sequence B[n] of integers.

Determine the starting and ending numbers

of the array elements that make up the

sequence for which the values of the

elements are equal to each other.}

//Const Value

Const

N = 20;

//Var Value

Var

Mas: array [1..N] of Integer;

Flag, Stop: Boolean;

CurrentValue, PreviouValue, NextValue: Integer;

First, Last: Integer;

Counter: Integer;

I: Integer;

Value: String;

Error: Integer;

// Mas - array [1..N] of Integer

// Flag – Variable to check

// Stop - Variable to check

// CurrentValue – Value of Element Mas[I]

// PreviouValue – Value of Element Mas[I – 1]

// NextValue – Value of Element Mas[I + 1]

// First – Value of first element in the sequence

// Last - Value of last element in the sequence

// Counter – Counter of the cycle

// I – Counter of the cycle

// Value – Value for Val

// Error – Value for Val

Begin

//Start of the sycle C1

for i := 1 to N do

Begin

//Start of the sycle C2

Repeat

//checking the correctness of the input

WriteLn('Type Mas[', i, ']: ');

ReadLn(Value);

val(Value, Mas[i], Error);

If (Error <> 0) then

Writeln('Incorrect value');

//End of the cycle C2

Until (Error = 0);

//End of the cycle C1

End;

i := 2;

Flag := False;

Stop := False;

//Start of the cycle A1

While (i <= Length(Mas) - 1) do

Begin

//Start of the cycle A2

while ((Flag = False) and (i <= Length(Mas))) do

Begin

//Definition of the elements to be compared

CurrentValue := Mas[i];

NextValue := Mas[i + 1];

PreviouValue := Mas[i - 1];

//Finding the last element of the sequence

if (CurrentValue = PreviouValue) and

(Stop = False) then

Begin

First := i - 1;

Stop := True;

End;

if ((CurrentValue <> NextValue) or

(CurrentValue <> PreviouValue)) and

(Stop = True) then

Begin

if CurrentValue <> NextValue then

Begin

Last := i;

End

else

if CurrentValue <> PreviouValue then

Begin

Last := i - 1;

End;

//Condition for starting the search

//for new sequences

if (PreviouValue <> CurrentValue)

or (NextValue <> CurrentValue) then

Begin

Flag := True;

End;

End;

i := i + 1;

//End of the cycle A2

End;

Flag := False;

Stop := False;

Counter := First;

//Output of a sequence and its start and end //numbers

Write('A sequence of equal numbers: ');

while COunter <= Last do

Begin

Write(Mas[First]);

Counter := Counter + 1;

End;

WriteLn(#10 ,'Number of First in the initial

array = ', First, #10, 'Number of Last in the

initial array = ', Last, #10);

//End of the cycle A1

End;

ReadLn;

End.

Приложение Б

(обязательное)

Тестовые наборы

Тестовая ситуация 1 для проверки ввода данных

Тест 1

Тестовая ситуация для проверки, сообщит ли программа о некорректном вводе, если введен элемент,не являющийся числом.

Исходные данные: Mas[1]: 1

Mas[2]: g

Ожидаемый результат: Incorrect value.

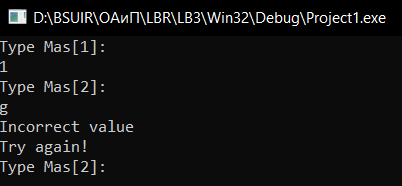
Полученный результат:

Рисунок 5 – Полученный результат, тест 1

Тест 2

Тестовая ситуация для проверки, сообщит ли программа о некорректном вводе, если введен элемент массива больше допустимого значения Integer.

Исходные данные: Mas[1]: 99999999999

Ожидаемый результат: Incorrect value.

Полученный результат:

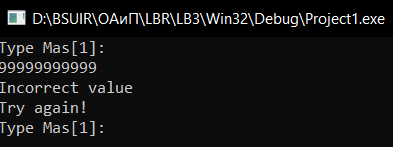


Рисунок 6 – Полученный результат, тест 2

Тест 3

Тестовая ситуация для проверки, сообщит ли программа о некорректном вводе, если введен не целочисленный элемент массива.

Исходные данные: Mas[1]: 1.1

Ожидаемый результат: Incorrect value.

Полученный результат:

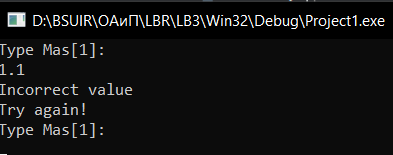


Рисунок 7 – Полученный результат, тест 3

Приложение В

(обязательное)

Исходный код программы (по постоновке 2)

Массив заполняется с использованием генератора чисел

Program LBR\_3;

{Enter n and a sequence B[n] of integers.

Determine the starting and ending numbers

of the array elements that make up the

sequence for which the values of the

elements are equal to each other.}

//Const Value

Const

N = 20;

//Var Value

Var

Mas: array [1..N] of Integer;

Flag, Stop: Boolean;

CurrentValue, PreviouValue, NextValue: Integer;

First, Last: Integer;

Counter: Integer;

I: Integer;

Value: String;

Error: Integer;

// Mas - array [1..N] of Integer

// Flag – Variable to check

// Stop - Variable to check

// CurrentValue – Value of Element Mas[I]

// PreviouValue – Value of Element Mas[I – 1]

// NextValue – Value of Element Mas[I + 1]

// First – Value of first element in the sequence

// Last - Value of last element in the sequence

// Counter – Counter of the cycle

// I – Counter of the cycle

// Value – Value for Val

// Error – Value for Val

Begin

//Start of the sycle C1

for i := 1 to N do

Begin

Mas[N] := Random(10);

//End of the cycle C1

End;

i := 2;

Flag := False;

Stop := False;

//Start of the cycle A1

While (i <= Length(Mas) - 1) do

Begin

//Start of the cycle A2

while ((Flag = False) and (i <= Length(Mas))) do

Begin

//Definition of the elements to be compared

CurrentValue := Mas[i];

NextValue := Mas[i + 1];

PreviouValue := Mas[i - 1];

//Finding the last element of the sequence

if (CurrentValue = PreviouValue) and

(Stop = False) then

Begin

First := i - 1;

Stop := True;

End;

if ((CurrentValue <> NextValue) or

(CurrentValue <> PreviouValue)) and

(Stop = True) then

Begin

if CurrentValue <> NextValue then

Begin

Last := i;

End

else

if CurrentValue <> PreviouValue then

Begin

Last := i - 1;

End;

//Condition for starting the search

//for new sequences

if (PreviouValue <> CurrentValue)

or (NextValue <> CurrentValue) then

Begin

Flag := True;

End;

End;

i := i + 1;

//End of the cycle A2

End;

Flag := False;

Stop := False;

Counter := First;

//Output of a sequence and its start and end //numbers

Write('A sequence of equal numbers: ');

while COunter <= Last do

Begin

Write(Mas[First]);

Counter := Counter + 1;

End;

WriteLn(#10 ,'Number of First in the initial

array = ', First, #10, 'Number of Last in the

initial array = ', Last, #10);

//End of the cycle A1

End;

ReadLn;

End.

Приложение Г

(обязательное)

Исходный код программы (по постоновке 3)

Массив заполняется с использованием типизированной константы-массив

Program LBR\_3;

{Enter n and a sequence B[n] of integers.

Determine the starting and ending numbers

of the array elements that make up the

sequence for which the values of the

elements are equal to each other.}

//Const Value

Const

N = 20;

S: array [1..N] of Integer = (1, 2, 2, 2, 3, 4,

5, 5, 5, 5, 6, 4,

4, 4, 7, 8, 1, 4,

1, 4);

//Var Value

Var

Mas: array [1..N] of Integer;

Flag, Stop: Boolean;

CurrentValue, PreviouValue, NextValue: Integer;

First, Last: Integer;

Counter: Integer;

I: Integer;

Value: String;

Error: Integer;

// Mas - array [1..N] of Integer

// Flag – Variable to check

// Stop - Variable to check

// CurrentValue – Value of Element Mas[I]

// PreviouValue – Value of Element Mas[I – 1]

// NextValue – Value of Element Mas[I + 1]

// First – Value of first element in the sequence

// Last - Value of last element in the sequence

// Counter – Counter of the cycle

// I – Counter of the cycle

// Value – Value for Val

// Error – Value for Val

Begin

//Start of the cycle C1

For I := 1 to N do

Begin

Mas[i] := S[i]

//End of the cycle C1

End;

i := 2;

Flag := False;

Stop := False;

//Start of the cycle A1

While (i <= Length(Mas) - 1) do

Begin

//Start of the cycle A2

while ((Flag = False) and (i <= Length(Mas))) do

Begin

//Definition of the elements to be compared

CurrentValue := Mas[i];

NextValue := Mas[i + 1];

PreviouValue := Mas[i - 1];

//Finding the last element of the sequence

if (CurrentValue = PreviouValue) and

(Stop = False) then

Begin

First := i - 1;

Stop := True;

End;

if ((CurrentValue <> NextValue) or

(CurrentValue <> PreviouValue)) and

(Stop = True) then

Begin

if CurrentValue <> NextValue then

Begin

Last := i;

End

else

if CurrentValue <> PreviouValue then

Begin

Last := i - 1;

End;

//Condition for starting the search

//for new sequences

if (PreviouValue <> CurrentValue)

or (NextValue <> CurrentValue) then

Begin

Flag := True;

End;

End;

i := i + 1;

//End of the cycle A2

End;

Flag := False;

Stop := False;

Counter := First;

//Output of a sequence and its start and end //numbers

Write('A sequence of equal numbers: ');

while COunter <= Last do

Begin

Write(Mas[First]);

Counter := Counter + 1;

End;

WriteLn(#10 ,'Number of First in the initial

array = ', First, #10, 'Number of Last in the

initial array = ', Last, #10);

//End of the cycle A1

End;

ReadLn;

End.

Приложение Д

(обязательное)

Исходный код программы (по постоновке 4)

Массив заполняется с использованием типизированной переменной-массив

Program LBR\_3;

{Enter n and a sequence B[n] of integers.

Determine the starting and ending numbers

of the array elements that make up the

sequence for which the values of the

elements are equal to each other.}

//Const Value

Const

N = 20;

//Var Value

Var

Mas: array [1..N] of Integer = (1, 2, 2, 2, 3, 4,

5, 5, 5, 5, 6, 4,

4, 4, 7, 8, 1, 4,

1, 4);

Flag, Stop: Boolean;

CurrentValue, PreviouValue, NextValue: Integer;

First, Last: Integer;

Counter: Integer;

I: Integer;

Value: String;

Error: Integer;

// Mas - array [1..N] of Integer

// Flag – Variable to check

// Stop - Variable to check

// CurrentValue – Value of Element Mas[I]

// PreviouValue – Value of Element Mas[I – 1]

// NextValue – Value of Element Mas[I + 1]

// First – Value of first element in the sequence

// Last - Value of last element in the sequence

// Counter – Counter of the cycle

// I – Counter of the cycle

// Value – Value for Val

// Error – Value for Val

Begin

i := 2;

Flag := False;

Stop := False;

//Start of the cycle A1

While (i <= Length(Mas) - 1) do

Begin

//Start of the cycle A2

while ((Flag = False) and (i <= Length(Mas))) do

Begin

//Definition of the elements to be compared

CurrentValue := Mas[i];

NextValue := Mas[i + 1];

PreviouValue := Mas[i - 1];

//Finding the last element of the sequence

if (CurrentValue = PreviouValue) and

(Stop = False) then

Begin

First := i - 1;

Stop := True;

End;

if ((CurrentValue <> NextValue) or

(CurrentValue <> PreviouValue)) and

(Stop = True) then

Begin

if CurrentValue <> NextValue then

Begin

Last := i;

End

else

if CurrentValue <> PreviouValue then

Begin

Last := i - 1;

End;

//Condition for starting the search

//for new sequences

if (PreviouValue <> CurrentValue)

or (NextValue <> CurrentValue) then

Begin

Flag := True;

End;

End;

i := i + 1;

//End of the cycle A2

End;

Flag := False;

Stop := False;

Counter := First;

//Output of a sequence and its start and end //numbers

Write('A sequence of equal numbers: ');

while COunter <= Last do

Begin

Write(Mas[First]);

Counter := Counter + 1;

End;

WriteLn(#10 ,'Number of First in the initial

array = ', First, #10, 'Number of Last in the

initial array = ', Last, #10);

//End of the cycle A1

End;

ReadLn;

End.