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: Одномерные массивы

Выполнил

студент: гр. 151003 Барановский Р.А.

Проверил: Фадеева Е.П.

Минск 2021

содержание

[1 Постановка задачи 3](#_Toc86596640)

[2 Текстовый алгоритм решения задачи 4](#_Toc86596641)

[3 Структура данных 6](#_Toc86596642)

[4 Схема алгоритма решения задачи по Гост 19.701-90 7](#_Toc86596643)

[5 Результаты расчетов 8](#_Toc86596644)

[Приложение А1 9](#_Toc86596645)

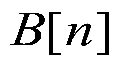
[Приложение А2 12](#_Toc86596646)

[Приложение А3 14](#_Toc86596647)

[Приложение А4 16](#_Toc86596648)

[Приложение Б 18](#_Toc86596649)

# Постановка задачи

 Из массива  исключить все элементы, у которых есть равные, оставив при этом последние вхождения элементов.

# Текстовый алгоритм решения задачи

Таблица – Алгоритм решения

|  |  |
| --- | --- |
| Номер  шага | Назначение шага |
| 1 | Const N = 10 |
| 2 | Error := 1 |
| 3 | i := 1 |
| 4 | Начало цикла А1. Проверка выполнения условия i <= N. Если условие истинно, то идти к шагу 5, иначе к шагу 16 |
| 5 | Начало цикла А2. Идти к шагу 6 |
| 6 | Ввод BStr[i] |
| 7 | Error := 1 |
| 8 | Проверка выполнения условия BStr[i] – вещественное число. Если условие истинно, то идти шагу 9, иначе к шагу 11 |
| 9 | B[i] = BStr[i] |
| 10 | Error := 0 |
| 11 | Проверка выполнения условия Error < > 0. Если условие истинно, то идти к шагу 12, иначе к шагу 13 |
| 12 | Вывод ‘Пожалуйста, введите вещественное число’ |
| 13 | Конец цикла А2. Проверка выполнения условия Error = 0. Если условие истинно, то идти к шагу 14, иначе к шагу 5. |
| 14 | i := i + 1 |
| 15 | Конец цикла А1. Идти к шагу 4 |
| 16 | i := 1 |
| 17 | Начало цикла А3. Проверка выполнения условия i <= N. Если условие истинно, то идти к шагу 18, иначе к шагу 21 |
| 18 | Вывод B[i] |
| 19 | i := i + 1 |
| 20 | Конец цикла А3. Идти к шагу 17 |
| 21 | Size := N |
| 22 | i := 0 |
| 23 | Начало цикла А4. Проверка выполнения условия i < Size. Если условие истинно, то идти к шагу 24, иначе к шагу 38 |
| 24 | i := i + 1 |
| 25 | j := i |
| 26 | Начало цикла А5. Проверка выполнения условия j < Size. Если условие истинно, то идти к шагу 27, иначе к шагу 37 |
| 27 | j := j + 1 |
| 28 | Проверка выполнения условия B[i] = B[j]. Если условие истинно, то идти к шагу 29, иначе к шагу 36 |
|  |  |
| 29 | k := j |
|  |  |
| 30 | Начало цикла А6. Проверка выполнения условия k <= Size – 1. Если условие истинно, то идти к шагу 31, иначе к шагу 34 |
| 31 | B[k] := B[k + 1] |
| 32 | k := k + 1 |
| 33 | Конец цикла А6. Идти к шагу 30 |
| 34 | Size := Size - 1 |
| 35 | j := i |
| 36 | Конец цикла А5. Идти к шагу 26 |
| 37 | Конец цикла А4. Идти к шагу 23 |
| 38 | i := 1 |
| 39 | Начало цикла А7. Проверка выполнения условия i <= Size. Если условие истинно, то идти к шагу 40, иначе к шагу 43 |
| 40 | Вывод B[i] |
| 41 | i := i + 1 |
| 42 | Конец цикла А7. Идти к шагу 39 |
| 43 | Останов. |

# Структура данных

Таблица - Данные

|  |  |  |
| --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение |
| N | Const | Начальный размер массива |
| B | Array of real | Массив |
| BStr | Array of String | Проверка ошибок ввода |
| i | Integer | Счетчик 1 |
| j | Integer | Счетчик 2 |
| k | Integer | Счетчик 3 |
| Error | Integer | Переменная, принимающая значение 1, если пользователь допустил ошибку при вводе. 0, если ошибок допущено не было |
| Size | Integer | Конечный размер массива |

# Схема алгоритма решения задачи по Гост 19.701-90

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

# Результаты расчетов

Вследствие результатов программы на экран выводятся следующие результаты расчетов:

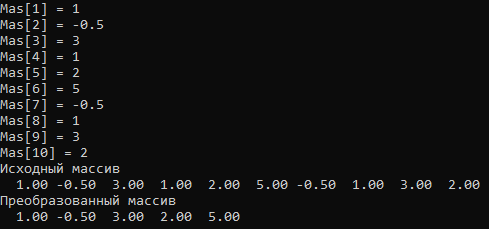


Рисунок 2 – Результаты расчетов 1 программы

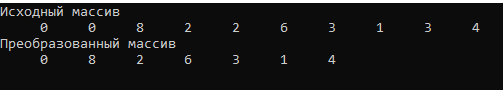


Рисунок 3 – Результаты расчетов 2 программы

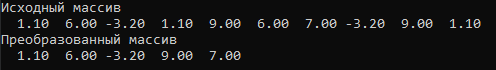


Рисунок 4 – Результаты расчетов 3 программы

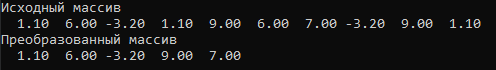


Рисунок 5 - Результаты расчетов 4 программы

Приложение А1

(обязательное)

Исходный код программы

program Lab3Exp1;

//Program exclude repeating elements

//Console app

{$APPTYPE CONSOLE}

{$R \*.res}

//Modules declaration

uses

System.SysUtils;

//Constants declaration

Const

N = 10;

//Arrays

Type

Mas = array[1..N] of real;

MasStr = array [1..N] of String;

//Variables declaration

Var

B: Mas;

BStr: MasStr;

i, j, k, Size, Error: integer;

{ Mas - array

MasStr - mistakes checker

i - counter 1

j - counter 2

k - counter 3

Size - current size of array

Error - variable which is 1 if user made

a mistake and 0 if not }

//Start the program

Begin

//Set Error to the original value

Error := 1;

//Inputting array elements

for i:=1 to n do

begin

//Protection from mistakes

repeat

write('B[',i,'] = ');

readln(BStr[i]);

val(BStr[i], B[i], Error);

//If mistake happened warn about it

if Error <> 0 then

writeln('Пожалуйста, введите вещественное число');

//If Error = 1 repeat the cycle

Until

Error = 0;

end;

//Output original array

writeln('Исходный массив');

for i:=1 to N do

write(B[i]:6:2);

//Remember the original size of array

Size:=N;

//Set i to the original value

i:=0;

//Entering the cycle

while i < Size do

begin

//Increment i

inc(i);

//Make j equals to i

j:=i;

//Entering the cycle

While j<Size do

begin

//Increment j

inc(j);

//If repeated element was found

if B[i]=B[j] then

begin

For k:=j to Size-1 do

//Move elements to the left

B[k]:=B[k+1];

//Decrease size of array

dec(Size);

//If number repeated more than once

j := i;

end;

end;

end;

writeln;

//Output new array

writeln('Преобразованный массив');

for i:=1 to Size do

write(B[i]:6:2);

//Final operations

readln;

End.

Приложение А2

(обязательное)

Исходный код программы

program Lab3Exp2;

//Program exclude repeating elements

//Console app

{$APPTYPE CONSOLE}

{$R \*.res}

//Modules declaration

uses

System.SysUtils;

//Constants declaration

Const

N = 10;

//Array

Type

Mas = array[1..N] of integer;

//Variables declaration

Var

B: Mas;

i, j, k, Size: integer;

{ Mas - array

i - counter 1

j - counter 2

k - counter 3

Size - current size of array }

//Start the program

Begin

//Generate random elements of array

Randomize;

for i := 1 to N do

B[i]:=Random(10);

//Output original array

writeln('Исходный массив');

for i:=1 to N do

write(B[i]:6);

//Remember the original size of array

Size:=N;

//Set i to the original value

i:=0;

//Entering the cycle

while i<Size do

begin

//Increment i

inc(i);

//Make j equals to i

j:=i;

//Entering the cycle

While j<Size do

begin

//Increment j

inc(j);

//If repeated element was found

if B[i]=B[j] then

begin

For k:=j to Size-1 do

//Move elements to the left

B[k]:=B[k+1];

//Decrease size of array

dec(Size);

//If number repeated more than once

j := i;

end;

end;

end;

writeln;

//Output new array

writeln('Преобразованный массив');

for i:=1 to Size do

write(B[i]:6);

//Final operations

readln;

End.

Приложение А3

(обязательное)

Исходный код программы

program Lab3Exp3;

//Program exclude repeating elements

//Console app

{$APPTYPE CONSOLE}

{$R \*.res}

//Modules declaration

uses

System.SysUtils;

//Constants declaration

Сonst

N = 10;

Mas: array[1..N] of real = (1.1,6,-3.2,1.1,9,6,7,

-3.2,9,1.1);

//Variables declaration

Var

i, j, k, Size: integer;

B: array[1..N] of real;

{ B - array

i - counter 1

j - counter 2

k - counter 3

Size - current size of array }

//Start the program

Begin

//Output original array

writeln('Исходный массив');

for i:=1 to N do

write(Mas[i]:6:2);

//Fill array B

for i := 1 to N do

B[i] := Mas[i];

//Remember the original size of array

Size:=N;

//Set i to the original value

i:=0;

//Entering the cycle

while i < Size do

begin

//Increment i

inc(i);

//Make j equals to i

j:=i;

//Entering the cycle

While j<Size do

begin

//Increment j

inc(j);

//If repeated element was found

if B[i]=B[j] then

begin

For k:=j to Size-1 do

//Move elements to the left

B[k]:=B[k+1];

//Decrease size of array

dec(Size);

//If number repeated more than once

j := i;

end;

end;

end;

writeln;

//Output new array

writeln('Преобразованный массив');

for i:=1 to Size do

write(B[i]:6:2);

//Final operations

readln;

End.

Приложение А4

(обязательное)

Исходный код программы

program Lab3Exp3;

//Program exclude repeating elements

//Console app

{$APPTYPE CONSOLE}

{$R \*.res}

//Modules declaration

uses

System.SysUtils;

//Constants declaration

Сonst

N = 10;

//Variables declaration

Var

i, j, k, Size: integer;

B:array[1..N] of real = (1.1,6,-3.2,1.1,9,6,7,

-3.2,9,1.1);

{ B - array

i - counter 1

j - counter 2

k - counter 3

Size - current size of array }

//Start the program

Begin

//Output original array

writeln('Исходный массив');

for i:=1 to N do

write(B[i]:6:2);

//Remember the original size of array

Size:=N;

//Set i to the original value

i:=0;

//Entering the cycle

while i < Size do

begin

//Increment i

inc(i);

//Make j equals to i

j:=i;

//Entering the cycle

While j<Size do

begin

//Increment j

inc(j);

//If repeated element was found

if B[i]=B[j] then

begin

For k:=j to Size-1 do

//Move elements to the left

B[k]:=B[k+1];

//Decrease size of array

dec(Size);

//If number repeated more than once

j := i;

end;

end;

end;

writeln;

//Output new array

writeln('Преобразованный массив');

for i:=1 to Size do

write(B[i]:6:2);

//Final operations

readln;

End.

Приложение Б

(обязательное)

Тестовые наборы

Таблица – Тестовые наборы

|  |  |  |
| --- | --- | --- |
| Тест | Исходные данные и ожидаемый результат | Полученный результат |
| 1 | Исходные данные: B[i] := Random (10)  Ожидаемый результат: программа уберет повторяющиеся элементы |  |
| 2 | Исходные данные: B[i] := Random (10) – 5  Ожидаемый результат: программа уберет повторяющиеся элементы |  |
| 3 | Исходные данные: B[i] := Random(10) – 10  Ожидаемый результат: программа уберет повторяющиеся элементы |  |
| 4 | Исходные данные: B[i] := Random(100)  Ожидаемый результат: программа уберет повторяющиеся элементы |  |
|  |  |  |
| 5 | Исходные данные: BStr[i] = ‘hooray’  Ожидаемый результат:  ‘Пожалуйста, введите вещественное число’ |  |