Министерство образования Республики Беларусь

Учреждение образования

БелорусскиЙ государственный университет

информатики и радиоэлектроники

Факультет компьютерных систем и сетей

Кафедра программного обеспечения информационных технологий

|  |
| --- |
|  |
|  |

# оТЧЕТ

по лабораторной работе

на тему:

Одномерные массивы

|  |  |  |
| --- | --- | --- |
| Выполнил  Студент гр. 451001 |  | Д. Д. Хренков |
| Проверил |  | Асс. Е.Е. Фадеева |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Минск, 2024

1. Теоретические сведения по теме лабораторной работы

Массив - упорядоченная совокупность однотипных элементов , имеющих общее имя.

1)Такое имя называют полной переменной;

2)Тип элементов - базовый тип;

Обращение к элементам массива происходит по индексам.

Индексное выражение – это выражение скалярного перенумерованного типа. Количество индексных выражений определяется количеством измерений массива.

Тип индекса:

1)Любой перенумерованный тип;

2)Количество элементов массива определяется количеством

возможных значений этого типа;

- В качестве индекса может быть указано любое выражение , имеющее тот же тип , что и индекс.

- В памяти массив представляет собой последовательно расположенные друг за другом переменные базового

типа.

Выход за границы массива - грубейшая ошибка!!!

- Может приводить к повреждению данных .

- В некоторых языках приводит к неопределенному поведению.

- Часто такая ошибка долго не проявляется и обнаруживается при внесении значительных изменений в программу.

Элементами массива могут быть массивы.

Существует две формы записи:

- сокращенная ;

- полная.

Над элементами массивов определены те же операции, что и над их базовыми типами .

Индексированная переменная - элемент массива.

- Ведет себя , как и обычная переменная базового типа.

Над полными переменными(полная переменная – сам массив) не определено никаких операций.

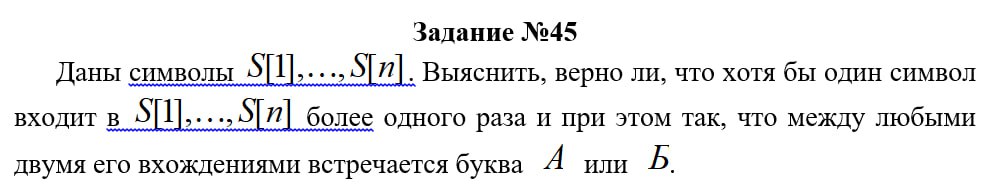
Полные переменные могут использоваться в операторе присваивания. Условия:

- Массивы должны быть одного и того же типа.

- При этом происходит копирование всех элементов.

Для инициализации массива в языке Паскаль могут быть использованы типизованные константы типа массив.

Задание на лабораторную работу

* 1. Постановка задачи
  2. Эскиз ожидаемого результата

Введите символ №1: C

Введите символ №2: A

Введите символ №3: C

Введите символ №4: B

Введите символ №5: C

Данная последовательность СОДЕРЖИТ специальный символ

1. Выполнение
   1. Разработка алгоритма

Таблица 3.1 используемые идентификаторы

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Имя идентификатора | Назначение | Тип идентификатора | Начальное значение | Закон изменения | Имя цикла, в котором происходит изменение переменной |
| n | Максимальное количество символов в ответе | integer | 10 | const | - |
| symbolResult | Хранит, есть ли специальный символ в последовательности | Boolean | - | symbolResult := hasSymbol(s) | - |
| S | Вводимая последовательность символов | tmas | - | s[I] := inputChar | ЦВЗ |
| i | Переменная-счетчик | integer | - | i := i + 1 | ЦВЗ, ЦПi |
| j | Переменная-счетчик | integer | - | j := j + 1 | ЦПj |
| k | Переменная счетчик | integer | - | k := k - 1 | ЦПk |
| nextAppear | Хранит новый индекс появления повторяющегося элемента | integer | i | nextAppear := j | ЦПj |
| inputChar | Хранит введённый пользователем символ | Char | - | readln(inputChar) | ЦВЗ |
| resultForSymbol | Хранит соблюдение условия для специального символа за конкретным символом | Boolean | fasle | resultForSymbol := true resultForSymbol := false | ЦПk |
| resForArray | Хранит соблюдение условия для специального символа за последовательностью | boolean | false | resForArray := true resForArray := false | ЦПi |

Рисунок 3.1 – Схема работы программы

* 1. Текст программы и его описание

program lab4;

uses

SysUtils;

const

n = 8;

type

TMAS = array [0 .. n - 1] of char;

var

S: TMAS;

symbolResult: Boolean;

procedure GenerateRandomChars(var S: TMAS);

var

i: Integer;

randomChar: char;

begin

Randomize;

write('Сгенерированная последовательность: ');

for i := 0 to n - 1 do

begin

// Генерируем случайную букву латинского алфавита в верхнем регистре

randomChar := Chr(Random(26) + 65);

write(randomChar);

S[i] := randomChar;

end;

writeln;

end;

procedure EnterChars(var S: TMAS);

var

i: Integer;

inputChar: char;

begin

for i := 0 to n - 1 do

begin

write('Введите символ №', i + 1, ': ');

readln(inputChar);

S[i] := inputChar;

end;

end;

function HasSymbol(const S: TMAS): Boolean;

var

i, j, k: Integer;

nextAppear: Integer;

resultForSymbol, resultForArr: Boolean;

begin

resultForArr := False;

for i := 0 to n - 2 do

begin

nextAppear := i;

resultForSymbol := False;

for j := i + 1 to n - 1 do

begin

if (S[i] = S[j]) and (j - nextAppear > 1) then

begin

for k := nextAppear + 1 to j - 1 do

begin

if (S[k] = 'A') or (S[k] = 'B') then

begin

resultForSymbol := True;

Break;

end

else

resultForSymbol := False;

end;

nextAppear := j;

end;

end;

if resultForSymbol then

begin

resultForArr := True;

Break;

end;

end;

if resultForArr then

HasSymbol := True

else

HasSymbol := False;

end;

begin

GenerateRandomChars(S);

// EnterChars(s);

symbolResult := HasSymbol(S);

if symbolResult then

writeln('Данная последовательность СОДЕРЖИТ специальный символ')

else

writeln('Данная последовательность НЕ СОДЕРЖИТ специальный символ');

readln;

end.

* 1. Тестирование и отладка программы

Таблица 3.2 Прохождение тестов программы

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Специфика тестирования | Номер теста | Вводимые данные | Ожидаемый результат | Полученный результат |
| Проверка на правильность выполнения при содержании символа | 1 | CACBC | Введите символ №1: C  Введите символ №2: A  Введите символ №3: C  Введите символ №4: B  Введите символ №5: C  Данная последовательность СОДЕРЖИТ специальный символ | Как и ожидалось |
| Проверка на правильность выполнения при НЕ содержании символа | 2 | GHYTJ | Введите символ №1: G  Введите символ №2: H  Введите символ №3: Y  Введите символ №4: T  Введите символ №5: J  Данная последовательность НЕ СОДЕРЖИТ специальный символ | Как и ожидалось |
| Проверка при наличии только A или B | 3 | ABAAB | Введите символ №1: A  Введите символ №2: B  Введите символ №3: A  Введите символ №4: A  Введите символ №5: B  Данная последовательность СОДЕРЖИТ специальный символ | Как и ожидалось |
| Проверка при наличии A или B между вхождениями, но не всеми | 4 | NMANGNA | Введите символ №1: N  Введите символ №2: M  Введите символ №3: A  Введите символ №4: N  Введите символ №5: G  Введите символ №6: N  Введите символ №7: A  Данная последовательность НЕ СОДЕРЖИТ специальный символ | Как и ожидалось |

Анализ прохождения тестов, причины возникших ошибок, внесение изменений в алгоритм и текст программы

* 1. Итоговый текст программы

program lab4;

uses

SysUtils;

const

n = 8;

type

TMAS = array [0 .. n - 1] of char;

var

S: TMAS;

symbolResult: Boolean;

procedure GenerateRandomChars(var S: TMAS);

var

i: Integer;

randomChar: char;

begin

Randomize;

write('Сгенерированная последовательность: ');

for i := 0 to n - 1 do

begin

// Генерируем случайную букву латинского алфавита в верхнем регистре

randomChar := Chr(Random(26) + 65);

write(randomChar);

S[i] := randomChar;

end;

writeln;

end;

procedure EnterChars(var S: TMAS);

var

i: Integer;

inputChar: char;

begin

for i := 0 to n - 1 do

begin

write('Введите символ №', i + 1, ': ');

readln(inputChar);

S[i] := inputChar;

end;

end;

function HasSymbol(const S: TMAS): Boolean;

var

i, j, k: Integer;

nextAppear: Integer;

resultForSymbol, resultForArr: Boolean;

begin

resultForArr := False;

for i := 0 to n - 2 do

begin

nextAppear := i;

resultForSymbol := False;

for j := i + 1 to n - 1 do

begin

if (S[i] = S[j]) and (j - nextAppear > 1) then

begin

for k := nextAppear + 1 to j - 1 do

begin

if (S[k] = 'A') or (S[k] = 'B') then

begin

resultForSymbol := True;

Break;

end

else

resultForSymbol := False;

end;

nextAppear := j;

end;

end;

if resultForSymbol then

begin

resultForArr := True;

Break;

end;

end;

if resultForArr then

HasSymbol := True

else

HasSymbol := False;

end;

begin

GenerateRandomChars(S);

// EnterChars(s);

symbolResult := HasSymbol(S);

if symbolResult then

writeln('Данная последовательность СОДЕРЖИТ специальный символ')

else

writeln('Данная последовательность НЕ СОДЕРЖИТ специальный символ');

readln;

end.