**Задание**

Дан фрагмент программного кода в виде модуля, содержащего только блок обработки для некоторой задачи. Считывание данных и вывод результатов не приводятся.

Каждый модуль содержит несколько ошибок (около 5). Ошибки могут быть разных типов:

* синтаксические – не позволят программе запуститься (пропущен символ «;», неверное имя переменной или функции и т.п.);
* логические:
  + программа запускается, но вылетает при работе (выход за пределы массива, деление на 0 и т.п.);
  + программа работает, но выдает неверный результат (неверные скобки, перепутан + и – и т.п.);
  + программа выдает неверный результат или вылетает в исключительных случаях (например, при вводе отрицательной длины массива);
  + программа работает корректно, но в ней присутствует лишний код (неиспользуемые переменные, лишние действия и т.п.).

Необходимо выяснить, что должна делать программа и найти все ошибки. Все пояснения привести в комментариях.

Для тестирования программы создать отдельный программный модуль, к которому через uses необходимо подключить исходный модуль. Поскольку второй модуль создается только для отладки и тестирования, качественное оформление не требуется. Допускается ввод-вывод организовать как через пользовательский интерфейс, так и считыванием из файла. Допускается использовать любые готовые фрагменты кода из предыдущих задач или Интернета.

К сдаче предоставляются:

* исправленный исходный модуль с комментариями;
* модуль ввода-вывода для тестирования;
* набор тестов, подтверждающий корректность работы программы (продемонстрировать или скриншотами).

**Варианты**

|  |  |
| --- | --- |
| Андреюшкина Т. |  |
| Беликова Н. |  |
| Виноградов А. |  |
| Груднов Д. |  |
| Кандалова И. |  |
| Киселев М. |  |
| Крючкова В. |  |
| Кспоян Г. |  |
| Мишанин М. |  |
| Сербин И. |  |
| Федорова И. |  |
| Чугунова Е. |  |
| Яйков (Валиев) А. |  |

**Пример**

**Исходный код**

unit UExample;

interface

type

TRealArray = array of real;

TIntArray = array of integer;

function SelectRandom(a: TRealArray; n: integer): TRealArray;

implementation

function GenRandomArray(n: integer): TRealArray;

var i: integer;

begin

if n > 0 then

begin

SetLength(GenRandomArray, n);

for i := 0 to n - 1 do

GenRandomArray[i] := Random(100);

end;

end;

procedure SwapReal(var a, b: real);

var buf: real;

begin

buf := a;

a := b;

b := buf;

end;

procedure SwapInt(var a, b: integer);

var buf: integer;

begin

buf := a;

a := b;

b := buf;

end;

function SortArray(a: TRealArray): TIntArray;

var i, j, n: integer;

begin

n := Length(a);

SetLength(SortArray, n);

for i := 0 to n - 1 do

SortArray[i] := i;

for j := 1 to n - 1 do

for i := 0 to j do

if a[i] > a[i+1] then

begin

SwapReal(a[i],a[i+1]);

SwapInt(SortArray[i],SortArray[i+1]);

end;

end;

function SelectRandom(a: TRealArray; n: integer): TRealArray;

var b: TIntArray;

begin

m := Length(a);

if n > 0 and m > 0 then

begin

SetLength(SelectRandom, n);

if n = 1 then

SelectRandom[0] := a[Random[m]];

else

begin

b := SortArray(GenRandomArray(m));

for i := 0 to n-1 do

SelectRandom[i] := a[b[i]];

end;

end;

end;

BEGIN

Randomize;

END.

**Исправленный код с комментариями**

unit UExample;

interface

type

TRealArray = array of real; //массив целых чисел

TIntArray = array of integer; //массив целых чисел

//выбрать n случайных элементов из массива a

//количество элементов n должно быть положительным

//если n >= длины массива, будут возвращены все элементы

//возвращает выбранные элементы в виде нового массива

function SelectRandom(a: TRealArray; n: integer): TRealArray;

implementation

//сгенерировать массив случайных чисел 0..100 длиной n

//n должно быть положительным, иначе массив не будет сгенерирован

//возвращает сгенерированный массив

function GenRandomArray(n: integer): TRealArray;

var i: integer;

begin

if n > 0 then //проверка длины массива

begin

SetLength(GenRandomArray, n);

//генерация

for i := 0 to n - 1 do

GenRandomArray[i] := Random(100);

end;

end;

//поменять местами значения двух вещественных переменных a и b

procedure SwapReal(var a, b: real);

var buf: real;

begin

buf := a;

a := b;

b := buf;

end;

//поменять местами значения двух целых переменных a и b

procedure SwapInt(var a, b: integer);

var buf: integer;

begin

buf := a;

a := b;

b := buf;

end;

//сортировка массива a по возрастанию

//функция возвращает массив с индексами исходных элементов

//в отсортированном массиве

//например, [150, -5, 21] => [-5, 21, 150]

// [ 0, 1, 2] => [ 1, 2, 0]

function SortArray(a: TRealArray): TIntArray;

var i, j, n: integer;

begin

n := Length(a);

//заполняем массив индексов

SetLength(SortArray, n);

for i := 0 to n - 1 do

SortArray[i] := i;

//сортировка методом "пузырька"

for j := 1 to n - 2 do //ОШИБКА! при i = n - 1 элемент a[i+1] выходит за пределы массива

for i := 0 to j do

if a[i] > a[i+1] then

begin

//сортируется и исходный массив, и массив с индексами

SwapReal(a[i],a[i+1]);

SwapInt(SortArray[i],SortArray[i+1]);

end;

end;

//выбрать n случайных элементов из массива a

//функция возвращает массив выбранных элементов

function SelectRandom(a: TRealArray; n: integer): TRealArray;

var

b: TIntArray;

m, i: integer; //ОШИБКА! необъявленные переменные

begin

m := Length(a);

//количество элементов должно быть положительным

if (n > 0) and (m > 0) then //ОШИБКА! пропущены ()

//выбрать все элементы

if n >= m then //ОШИБКА! пропущен вариант, когда нуно выбрать все элементы массива

SelectRandom := copy(a, 0, m)

else

begin

SetLength(SelectRandom, n);

//выбрать только 1 элемент

if n = 1 then

//упрощенно, по случайному номеру

SelectRandom[0] := a[Random(m)] //ОШИБКА! [] вместо (), лишняя ;

else

//выбрать несколько элементов

begin

//генерируем и сортируем случайный массив

b := SortArray(GenRandomArray(m));

//отбираются n элементов с наименьшим случайным значением

//(первые n элементов в отсортированном массиве)

for i := 0 to n-1 do

SelectRandom[i] := a[b[i]];

end;

end;

end;

BEGIN

Randomize;

END.

**Модуль тестирования**

program Example;

uses UExample;

var a, b: TRealArray;

i, n, m: Integer;

BEGIN

Writeln('Введите длину массива:');

Readln(n);

Writeln('Введите элементы массива:');

SetLength(a, n);

for i := 0 to n-1 do

Readln(a[i]);

Writeln('Введите количество элементов, которые нужно выбрать:');

Readln(m);

b := SelectRandom(a, m);

Writeln('Результат:');

for i := 0 to Length(b)-1 do

Writeln(i+1, '. ', b[i]:0:2);

readln;

END.