Лабораторная работа №9. Разветвляющиеся вычислительные процессы. Оператор выбора.

Цель л/р:рассмотрение способов организации разветвляющихся вычислительных процессов средствами Lazarus и Free Pascal.

Оборудование: Word, Lazarus, Draw.io, ПК.

Задание 1.

1. 
2. Математическая модель:
3. Блок-схема:



1. Список идентификаторов:

|  |  |  |
| --- | --- | --- |
| Имя | Смысл | Тип |
| x | Вводимое пользователем пятиразрядное число | longint |

1. Код программы:

program Project1;

var x:longint;

begin

writeln('Enter x');

readln(x);

if (x mod 10 = x div 10000) and (x mod 100 div 10 = x div 1000 mod 10) then

writeln('Palindrome')

else writeln('Not palindrome');

readln();

end.

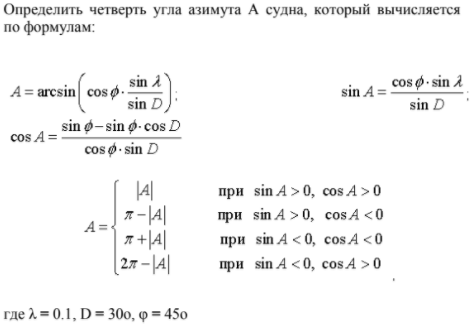
1. Результаты выполненной работы:



1. Анализ результатов вычисления:

Программа при запуске ожидает ввода положительного целого пятиразрядного числа. Далее происходит проверка равенства 0 и 4, 1 и 3 разрядов. В случае ложности условия на экран выводится сообщение ‘Not palindrome’, программа закрывается. Если условие истинно, на экран выводится сообщение ‘Palindrome’, после чего программа завершает работу.

Задание 2.

1. 
2. Математическая модель:
3. Блок-схема:



1. Список идентификаторов:

|  |  |  |
| --- | --- | --- |
| Имя | Смысл | Тип |
| l | Вводимое пользовтаелем значение | real |
| A | Значение азимута судна | real |
| cosA | Косинус азимута | real |
| sinA | Синус азимута | real |
| phi\_rad | phi в переводе в радианы | real |
| d\_rad | d в переводе в радианы | real |
| d | Вводимое пользователем значение | integer |
| phi | Вводимое пользователем значение | integer |

1. Код программы:

program Project2;

uses math, Unit1;

var l,A,cosA,sinA,phi\_rad,d\_rad:real;

d, phi:integer;

begin

l:=0.1;

d:=30;

phi:=45;

phi\_rad:=phi\*pi/180;

d\_rad:=d\*pi/180;

cosA:=((sin(phi\_rad)-(sin(phi\_rad)\*cos(d\_rad)))/(cos(phi\_rad)\*sin(d\_rad)));

sinA:=cos(phi\_rad)\*sin(l)/sin(d\_rad);

A:=arcsin(cos(phi\_rad)\*sin(l)/sin(d\_rad));

if sinA > 0 then begin

if cosA > 0 then begin

A:=abs(A);

writeln('First quarter');

end

else begin

A:=pi-abs(A);

writeln('Second quarter');

end;

end

else if cosA > 0 then begin

A:=2\*pi-abs(A);

writeln('Fourth quarter');

end

else begin

A:=pi+abs(A);

writeln('Third quarter');

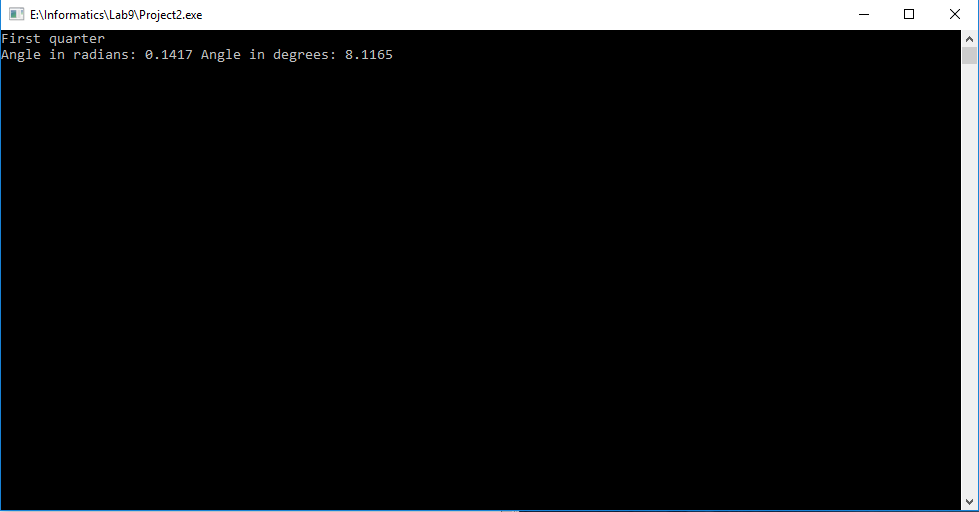
end;

writeln('Angle in radians: ',A:0:4,' Angle in degrees: ',A\*180/pi:0:4);

readln();

end.

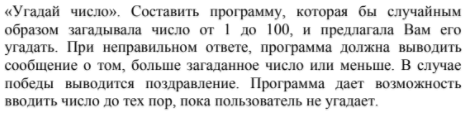
1. Результаты выполненной работы:



1. Анализ результатов вычисления:

При запуске программы происходит присвоение переменным значений заданных параметров, перевод переменных phi и d в радианы и запись полученных значений в phi\_rad и d\_rad соответственно. Далее расчитываются значения синуса и косинуса азимута, а также значение A, возвращаемое обратной тригонометрической функций arcsin. Далее осуществляется проверка значений sinA и cosA, организованная в соответствии с условием задания. При подстановке указанных в условии задания значений согласно расчётам азимут судна A принадлежит первой угловой четверти, также выводится на экран его значение в радианах и градусах.

Задание 3.

1. 
2. Математическая модель:
3. Блок-схема:



1. Список идентификаторов:

|  |  |  |
| --- | --- | --- |
| Имя | Смысл | Тип |
| x | Вводимое пользователем число | byte |
| y | Генерируемое процедурой random число, лежащее в диапазоне [1,100]. | byte |

1. Код программы:

program Project3;

var x,y:byte;

begin

randomize;

y:=random(100)+1;

repeat

writeln('Enter the number');

readln(x);

if x>y then

writeln('Your number is more than y');

if x<y then

writeln('Your number is less than y');

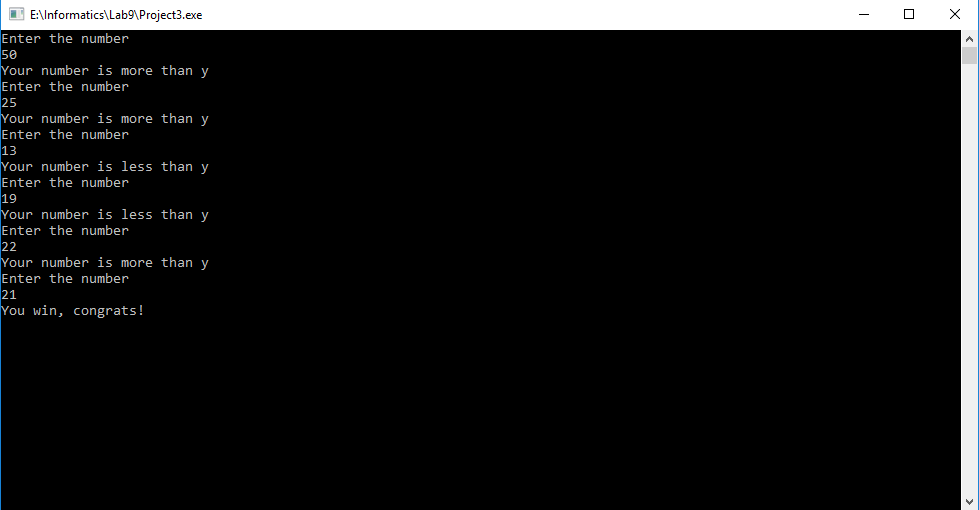
until x=y;

writeln('You win, congrats!');

readln();

end.

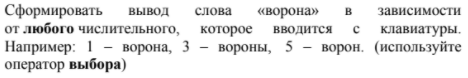
1. Результаты выполненной работы:



1. Анализ результатов вычисления:

При запуске программа инициализирует генератор случайных чисел, при помощи процедуры random() переменной y присваивается значение из диапазона [1,100]. Далее осуществялется вход в цикл, в теле которого пользователем вводится значение x, происходит последовательная проверка условий x>y и x<y, в случае истинности одного из них на экран выводится соответсвующее сообщение о значении большем/меньшем созданного программой. Выход из цикла происходит при выполнении условия x=y, на экране появляется сообщение с поздравлением для угадливого пользователя.

Задание 4.

1. 
2. Математическая модель:
3. Блок-схема:



1. Список идентификаторов:

|  |  |  |
| --- | --- | --- |
| Имя | Смысл | Тип |
| x | Вводимое пользователем кол-во ворон | word |

1. Код программы:

program Project4;

var x:word;

begin

writeln('Введите число ворон');

readln(x);

if (x mod 100 > 10) and (x mod 100 < 15) then

writeln(x, 'ворон')

else

case x mod 10 of

1:writeln(x,' ворона');

2..4:writeln(x,' вороны');

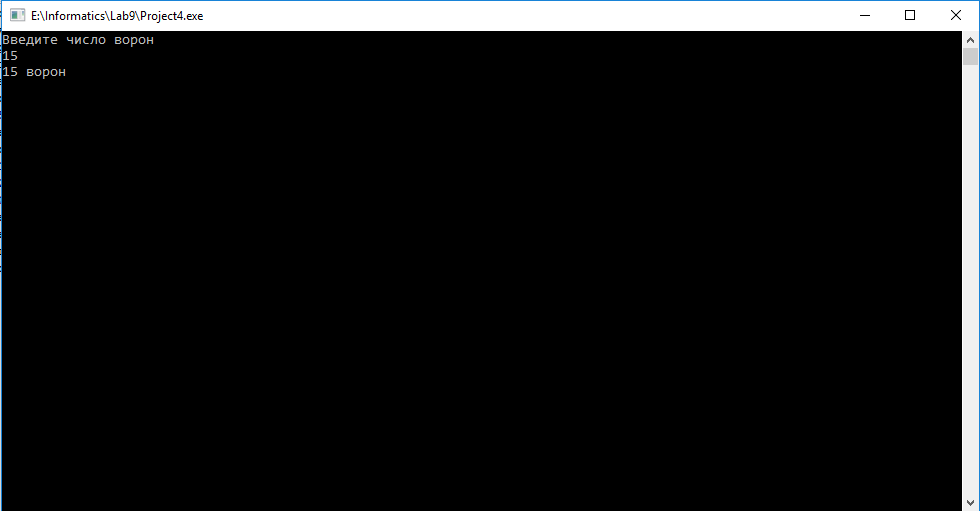
0,5..9:writeln(x,' ворон');

end;

readln();

end.

1. Результаты выполненной работы:



1. Анализ результатов вычисления:

При запуске программы осуществляется ввод пользователем количества ворон с клавиатуры. После записи значения в переменную x проверяются варианты x mod 100 = [11,14], т.к. запись в данном диапазоне не подчиняется правилам, распространяющимся на отбор остальных значений. В случае истинности условия на экран выводится кол-во ворон с дозаписью ‘ворон’, программа завершается.

В противном случае происходит переход к следующему оператору выбора case x mod 10 of, происходит выбор одной из трёх ветвей, соотвествующих поставленным требованиям. На экран выводится значение x с дозаписью ‘ворона’, ‘вороны’,’ворон’ для значений 1;2..4;0,5..9 соответственно. Программа заврешается.

Вывод:

Разветвляющиеся вычислительные процессы используются с целью выбора определенной ветви различных инструкций в зависимости от значения, поступающего в блок проверки условия. В частности, в ходе выполнения л/р была рассмотрена задача на определение четверти обратной тригонометрической функции. Во Free Pascal разветвляющиеся вычислительные процессы могут быть организованы при помощи конструкций

if …

else

и

case … of:

1:…

2:…

…

n:…