**III Тур олимпиады по программированию, 2016-2017 уч.год**

**Галляува Ильнара Робертовна, 11 класс,**

**образовательное учреждение: 452558, РБ Мечетлинский район, с.Алегазово, ул.Советская, 4, МОБУ СОШ с.Алегазово. e-mail gallyauva-i@yandex.ru**

Язык Pascal, выполнено в Lazarus IDE v1.2.4 Консольное приложение.

**Задание 1**

Разработайте программу, которая шифрует и дешифрует текст, используя шифр Цезаря для заданного сдвига номера каждой буквы алфавита на фиксированное число позиций от 1 до 20.

**Примеры входных и выходных данных:**

Входные данные

Шифрование - 1, дешифрование – 2: 1

Сдвиг (1-20): 1 Разработать программу для игры в морской бой на основе таблицы из одинаковых символов. Нумерация строк: сверху-вниз. Нумерация столбцов: слева-направо. Координаты размещения кораблей задаются прямым присваиванием значений в программе. Координаты выстрелов X и Y вводятся с клавиатуры. Результаты попаданий выводятся в виде таблицы символов. Придумать свое обозначение результатов попадания и не попадания в цель. Можно выбрать другой символ для обозначения исходного состояния таблицы из одинаковых символов. Игра завершается сообщением: Вы выиграли.

**Примеры входных и выходных данных:**

Начальная таблица символов:

\*\*\*\*\*

\*\*\*\*\*

\*\*\*\*\*

\*\*\*\*\*

1 выстрел

Ввод: x=3 и y=2

Вывод:

\*\*\*\*\*

\*\*н\*\*

\*\*\*\*\*

\*\*\*\*\*

2 выстрел

Ввод: x=5 и y=4

Вывод:

\*\*\*\*\*

\*\*н\*\*

\*\*\*\*п

\*\*\*\*\*

В этом примере выстрел без попадания: н. Выстрел с попаданием в корабль: п.

|  |
| --- |
| **Решение:**program project1;const n=5;//строк k=5;//столбцов var a:array [1..n,1..n] of byte; b:array [1..n,1..n] of char; x,y,k1,i,j,nn:integer; procedure post; var x1,y1:integer; begin x1:=random(n)+1; y1:=random(k)+1; if a[x1,y1]=0 then a[x1,y1]:=1 else post; end;begin k1:=0; for i:=1 to n do begin for j:=1 to k do begin a[i,j]:=0; b[i,j]:='.'; write(b[i,j]); end; writeln; end; for i:=1 to k do begin post; end; nn:=1; while k1<k do begin writeln(nn,' выстрел'); write('Ввод: '); write('x='); read(y); write('и y='); readln(x); writeln('Вывод: '); if a[x,y]=1 then begin b[x,y]:='+'; k1:=k1+1; end else b[x,y]:='-'; nn:=nn+1; for i:=1 to n do begin for j:=1 to n do write(b[i,j]); writeln; end; end; writeln; write('Вы выиграли. '); readln; readln;end.  |

**Задание 2**

Разработать программу для поиска разности числовых значений двух физических процессов, которая по модулю не превышает заданное число M. Один процесс изменяется каждую единицу времени N по формуле арифметической прогрессии A(n)=A(n-1)+D, другой – по формуле геометрической прогрессии B(n)=B(n-1)\*Q. Значения A(1), B(1) задаются прямым присваиванием в программе. Числовые значения D, Q и M задаются с клавиатуры. Программа выводит на экран числовые значения двух физических процессов в следующих случаях:

а) числовые значения двух физических процессов для n>1 имеют разность не превышающую по модулю число M;

б) числовые значения физических процессов за одну единицу времени до выхода из диапазона возможных значений переменных.

В программе можно использовать любые допустимые типы числовых переменных кроме типа byte. Данный тип переменной не позволяет использовать отрицательные числа и используется только для демонстрации примера входных и выходных данных.

Примечание:

 

**Примеры входных и выходных данных** с использованием в программе значений A(1)=1, B(1)=2 для числовых переменных типа byte:

 Ввод D, Q и M:

D Q M

100 2 200

Вывод N, A(n), B(n) и M=|A(n)-B(n)|;

N A(n) B(n) M

2 101 4 97

3 201 8 193

Завершение работы программы по значениям n=3, A(n)=201, B(n)=8, М=193.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Решение:**program project1; var a,b,d,q,m,n:integer;begin assign(input,'input.txt'); reset(input); assign(output,'output.txt'); rewrite(output); read(d,q,m); writeln('Вывод N, A(n), B(n) и M=|A(n)-B(n)|;' ); writeln('N A(n) B(n) M'); close(input); a:=1; b:=2; n:=2; b:=b\*q; a:=a+d\*(n-1); repeat if abs(a-b)<m then writeln(n,' ',a,' ',b,' ',abs(a-b)) else exit; b:=b\*q; a:=a+d\*(n-1); until abs(a-b)>=m; close(output);end.

|  |  |
| --- | --- |
| input.txt | output.txt |
| 100 2 200 | Вывод N, A(n), B(n) и M=|A(n)-B(n)|;N A(n) B(n) M2 101 4 972 201 8 193 |

 |

**Задание 3**

 Разработать программу, которая находит K производных степенной функции F=X^N от введенного с клавиатуры начального значения N до значения F`= 0.

**Примеры входных и выходных данных:**

Ввод N:

5

Вывод K, F, F`:

K F F`

1 x^5 5\*x^4

2 5\*x^4 20\*x^3

3 20\*x^3 60\*x^2

4 60\*x^2 120\*x

5 120\*x 120

6 120 0

Примечание:



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Решение:**program project1;var n,k,c,n1:integer;f,f1,ns,nc:string;begin assign(input,'input.txt'); reset(input); assign(output,'output.txt');rewrite(output); read(n); close(input); writeln( 'Вывод K, F, F`:'); writeln( 'K F F`'); c:=1; n1:=n; str(n1,ns); f1:='x^'+ns; for k:=1 to n+1 do begin f:=f1; c:=c\*n1; str(c,nc); n1:=n1-1; str(n1,ns); if c=0 then f1:='0' else if n1=1 then f1:=nc+'\*x' else if n1=0 then f1:=nc else f1:=nc+'\*x^'+ns; writeln(k,' ',f,' ',f1,' '); end;end.

|  |  |
| --- | --- |
| input.txt | output.txt |
| 5 | Вывод K, F, F`:K F F`1 x^5 5\*x^42 5\*x^4 20\*x^33 20\*x^3 60\*x^24 60\*x^2 120\*x5 120\*x 1206 120 0 |

Время выполнения 5 минВремя выполнения 18 мин |

**Задание 4**

Цилиндр из картона высотой H и диаметром D нужно обклеить листами цветной бумаги. Разработать программу для подсчета необходимого количества листов бумаги с фиксированными размером листа со сторонами А и В.

**Примеры входных и выходных данных:**

Ввод H, D, A, B:

H D A B

70 35 35 40

Вывод: 8 листов

|  |
| --- |
| **Решение:** |

**Задание 5**

 У исполнителя «Утроитель» две команды, которым присвоены номера:

1. Прибавь 1.

2. Умножь на 3.

Первая из них увеличивает число на 1, вторая – утраивает его.

Программа для «Утроителя» – это последовательность команд из цифр 1 и 2.

Вместо цифр для обозначения команд допускается использование других символов или их сочетаний.

 Разработать программу для вывода последовательностей не более чем из 14 команд, которые число Y1 преобразуют в число Y2? Сколько таких последовательностей - программ?

**Примеры входных и выходных данных:**

Ввод

Введите начальное число Y1: 5

Введите конечное число Y2: 15

Вывод

Последовательности команд:

 2

 1111111111

Программы преобразования из 5 в 15. Количество программ: 2.

|  |
| --- |
| **Решение:** **program project1;**const n=32767;vark0,k,i,j,z:longint;a:array[1..n] of integer;s:string;beginwrite('Введите начальное число y1: ');readln(k0);write('Введите конечное число y2: ');readln(k);writeln('Последовательности команд:');z:=0;a[1]:=k0;for i:=2 to n do beginif i mod 2=0 then a[i]:=a[i div 2]+1else a[i]:=a[i div 2]\*3;end;for i:=n downto 2 do beginif a[i]=k then beginz:=z+1;s:='';j:=i;while j<>1 do beginif j mod 2=0 then Insert('1',s,1) else Insert('2',s,1);j:=j div 2;end;writeln(s);end;end;write('Программы преобразования из ',k0,' в ',k,'. Количество программ: ',z);readln;readln;end. |

**Задание 6**

Разработать программу, которая проверяет правильность признаков делимости в интервале целых положительных чисел для делителя 11.

Число делится на 11, если сумма цифр, которые стоят на четных местах равна сумме цифр, стоящих на нечетных местах, либо отличается от неё на 11.

Проверка целочисленного деления по признакам в программе сравнивается с прямым делением с помощью оператора деления.

**Примеры входных и выходных данных:**

Ввод:

8338

Вывод:

Делится на 11. Результат: 758.

Результат подтверждается: 758.

|  |
| --- |
| **Решение:**program project1;varn,n1:longint;k1,k2:integer;beginreadln(n);n1:=n;while n>0 do begink1:=k1+(n mod 10);n:=n div 10;k2:=k2+(n mod 10);n:=n div 10;end;if (abs(k1-k2)=11) or (k1-k2=0) then beginwriteln('Делится на 11. Результат: ', n1 div 11,'.');if n1 mod 11=0 then writeln('Результат подтверждается: ', n1 div 11,'.');endelse begin writeln('Не делится на 11');if n1 mod 11<>0 then writeln('Результат подтверждается: число ', n1,' при делении на 11 даёт остаток ', n1 mod 11)end;readln;readln;end. |

**Задание 7**

 Предполагалось, что физический процесс изменяется линейно y=kx+b с параметрами прямой k и b. В результате эксперимента получено 10 пар значений:

k: 1.0, 2.0, 3.0, 5.0, 6.0, 7.0, 7.0, 8.0, 8.5, 9.0

b: 1.0, 1.5, 1.0, 2.0, 3.0, 2.0, 3.0, 4.0, 3.0, 5.0

 Разработайте программу, которая для каждой пары параметров находит среднеквадратичную ошибку. Определите с помощью программы такое сочетание k и b, при которых ошибка прогноза будет минимальной.

**Примеры входных и выходных данных:**

Пример ввода трех пар:

k: 2.0, 0.5, 0.8

b: 0.5, 0.3, 0.2

Вывод параметров и значений среднеквадратичной ошибки:

k= 2.0 , b = 0.5 , mserror= 104.125

**k= 0.5 , b = 0.3 , mserror= 0.78625**

k= 0.8 , b = 0.2 , mserror= 6.229

Минимальная ошибка прогноза: 0.78625 (k= 0.5 , b = 0.3).

**Решение:**