Коваленко П.И. гр. 3102

Лабораторная работа №8

Алгоритм игры:

1. Компьютер загадывает N-значное число (длина числа задаётся

пользователем при старте игры и, очевидно, не может превышать 10),

состоящее из неповторяющихся цифр (никакая цифра не может

присутствовать в числе дважды, число не может начинаться с нуля). Для

генерации случайного числа необходимо использовать методы

библиотечного класса Random.

2. Затем пользователь, пытаясь угадать загаданное число, вводит Nзначное число число.

3. Компьютер выводит сообщение о том, сколько цифр (коров) угадано, но

не расположено на своих местах, и сколько цифр (быков) угадано и

находится на своих местах.

4. После чего пользователь переходит к п.2. Раунды продолжаются до тех

пор, пока пользователь не отгадает загаданное число (т.е. получит

четыре “быка”).

Решение:

#include <iostream>

#include <stdlib.h>

#include <time.h>

//#define debag

using namespace std;

/// Функция для заполнения динамического массива

void fillDynamicArray (int \*inputDynamicArray, int numberElemets)

{

int variableDigit;

bool boolean = false;

inputDynamicArray[0] = rand() % 9 + 1; /// Заполняет первый элемент массива числом от 1 до 9

for (int i = 1; i < numberElemets; i++)

{

while(true)

{

variableDigit = rand() % 10; /// Заполняем переменное число

for (int j = 0; j < i; j++)

{

if(variableDigit == inputDynamicArray[j])

{

boolean = true;

break;

}

}

if (boolean == true)

boolean = false;

else

break;

}

inputDynamicArray[i] = variableDigit;

}

}

int strInDigit (char inputChar)

{

switch (inputChar)

{

case '0' : return 0;

case '1' : return 1;

case '2' : return 2;

case '3' : return 3;

case '4' : return 4;

case '5' : return 5;

case '6' : return 6;

case '7' : return 7;

case '8' : return 8;

case '9' : return 9;

}

}

int main()

{

setlocale(LC\_ALL, "Rus"); /// Локализуем русский язык

srand(time(NULL)); /// Создаём ключ генератора рандома

int n, cows, bulls;

char strNDigitNumber[10];

cout << "Введите n (1 < n < 10): ";

cin >> n;

int \*nDigitNumber = new int[n]; /// Выделяем память под динамический массив

fillDynamicArray(nDigitNumber, n);

int \*variableNDigitNumber = new int[n]; /// Выделяем память под динамический массив

while(bulls != n)

{

cows = 0;

bulls = 0;

#ifdef debag

for (int i = 0; i < n; i++)

cout << nDigitNumber[i] << " ";

#endif // debag

cout << "Введите ваше число: ";

cin >> strNDigitNumber;

for (int i = 0; i < n; i++)

{

variableNDigitNumber[i] = strInDigit(strNDigitNumber[i]);

}

for (int i = 0; i < n; i++)

{

for(int j = 0; j < n; j++)

{

if(variableNDigitNumber[i] == nDigitNumber[j] && i == j)

bulls++;

else if(variableNDigitNumber[i] == nDigitNumber[j])

cows++;

}

}

cout << "Коров = " << cows << endl;

cout << "Быков = " << bulls << endl;

}

cout << "Вы выйграли!" << endl;

delete [] nDigitNumber; /// Освобождаю память

delete [] variableNDigitNumber; /// Освобождаю память

return 0;

}

