

Министерство науки и высшего образования РФ
ФГБОУ ВПО «Пермский государственный национальный
исследовательский университет»
Кафедра прикладной математики и информатики

Индивидуальное задание №1
Игра “Быки и Коровы”
по дисциплине «Алгоритмизация и программирование»

Выполнил:

Студент 1-го курса механико-
математического ф-та
Шмырина В.О
Группа ПМИ-1,2 2021год

Проверил:

Старший преподаватель кафедры
прикладной математики и
информатики ПГНИУ,
_____ Шеина Т. Ю.

«__»_____ 2021 г.

Пермь 2021

Постановка задачи

Напишите игру “Быки и Коровы”

Условия задачи

Компьютер загадывает четырехзначное число, так чтобы все цифры числа были различные. Цель игрока – угадать это число. За каждый ход, отгадывающий называет свое четырехзначное число (пользователь его вводит с клавиатуры). Если цифра из названного числа есть в отгадываемом числе, но стоит не на своем месте, то эта ситуация называется «корова». Если цифра из названного числа есть в отгадываемом числе и стоит на своем месте, то эта ситуация называется «бык».

Определение идеи алгоритма, выбор методов решения и структур данных

Для решения задачи использовалась среда разработки Visual C++.

Код игры будет описан в виде функций, поскольку некоторые действия нам необходимо выполнять по несколько раз. Большинство функций будут иметь строковый тип. Это сделано для того, чтобы была возможность работать с каждой цифрой числа, при этом не выделяя из него отдельные цифры.

Программа состоит из четырех дополнительных функций и одной главной.

Дополнительные функции:

1. `string generation()` - не принимает параметров на вход, внутри происходит генерация случайного четырехзначного числа, которое имеет различные цифры. Функция возвращает сгенерированное число в главную функцию

2. `string vvod_number()` - не принимает параметров на вход, внутри происходит ввод данных пользователем. Функция возвращает полученные входные данные в функцию `bulls_cows()` только в том случае, когда произойдет проверка на корректность введенных данных в функции `checkNumber()`

3. `bool checkNumber(string num)` - функция проверки корректности введенного числа. На вход принимает число, введенное пользователем, если число введено некорректно, то программа попросит повторить ввод данных

4. `void bulls_cows(string digit_comp)` - функция, в которой происходит сравнение числа, сгенерированного компьютером, и введенного числа пользователем.

Происходит подсчет быков и коров. Если пользователь угадал число, то есть у него получилось 4 быка, то игра заканчивается, если нет, то программа снова попросит ввести вариант числа.

Главная функция:

`int main()` - функция, в которой происходит приветствие игрока и краткое объяснение правил. А также в ней вызывается функция, которая генерирует число и функция, которая ответственна за основную часть игры.

Тестирование программы

Программа была протестирована для 4-ех видов входных данных, для каждого из них ниже дано описание, представлен тест и результат прохождения данного теста программой.

- 1) Исходные данные не имеют ошибок; компьютер генерирует – число:

Результат, полученные при тестировании верный

РЕЗУЛЬТАТ, ПОЛУЧЕННЫЙ ПРИ ТЕСТИРОВАНИИ ПРОГРАММЫ:

```
число, загаданное компьютером 3091
Ваш номер хода: 1
Введите четырехзначное число с различными цифрами : 2091
Количество коров : 0
Количество быков: 3
Ваш номер хода: 2
Введите четырехзначное число с различными цифрами : 3091
Количество коров : 0
Количество быков: 4
Поздравляю! Вы отгадали число, задуманное компьютером за 2 хода
```

- 2) Исходные данные не имеют ошибок; на входе – число, с количеством символов, больше 4

Поскольку введенные данные не соответствуют требованиям игры, программа просит повторить ввод

РЕЗУЛЬТАТ, ПОЛУЧЕННЫЙ ПРИ ТЕСТИРОВАНИИ ПРОГРАММЫ:

```
число, загаданное компьютером 7023
Ваш номер хода: 1
Введите четырехзначное число с различными цифрами : 16803
Введите четырехзначное число с различными цифрами : 
```

- 3) Исходные данные содержат ошибку; на входе – число, с повторяющимися цифрами

Поскольку введенные данные не соответствуют требованиям игры, программа просит повторить ввод

РЕЗУЛЬТАТ, ПОЛУЧЕННЫЙ ПРИ ТЕСТИРОВАНИИ ПРОГРАММЫ:

```
число, загаданное компьютером 8537
Ваш номер хода: 1
Введите четырехзначное число с различными цифрами : 1147
Введите четырехзначное число с различными цифрами :
```

4) Исходные данные содержат ошибку: входные данные содержат букву.

Поскольку введенные данные не соответствуют требованиям игры, программа просит повторить ввод

РЕЗУЛЬТАТ, ПОЛУЧЕННЫЙ ПРИ ТЕСТИРОВАНИИ ПРОГРАММЫ:

```
число, загаданное компьютером 6497
Ваш номер хода: 1
Введите четырехзначное число с различными цифрами : 649э
Введите четырехзначное число с различными цифрами : _
```

Текст программы

Исходный файл программы прилагается к работе в электронном виде, ниже приводим его полностью.

```
#include <iostream>
#include <ctime>
#include <string>
```

```
using namespace std;
```

```
string generation() //генерирует 4-х значное число, которое необходимо угадать игроку
{
    srand(time(0));
    const int size = 4;
    int random[size];
    string comp = "";
    bool flag;
    for (int i = 0; i < size;)
    {
        flag = false;
        int newRandom = rand() % 10;
        for (int j = 0; j < i; j++)
        {
            if (random[j] == newRandom)
            {
                flag = true;
            }
        }
        if (!flag)
        {
            random[i] = newRandom;
            i++;
        }
    }
    for (int i = 0; i < 4; i++)
    {
        comp += to_string(random[i]);
    }
}
```

```

    // cout << "число, загаданное компьютером " << comp << endl;
    return comp;
}

```

```

bool checkNumber(string num) //проверка корректности числа, введенного игроком
{
    bool flag = true;
    if ((num.length() == 4))
    {
        for (int i = 0; i < num.length(); i++)
        {
            if (not(isdigit(num[i])))
            {
                flag = false;
            }
        }
        for (int j = 0; j < num.length() - 1; j++)
        {
            if (num[j] == num[j + 1]) flag = false;
        }
    }
    else flag = false;
    return flag;
}

```

```

string vvod_number() //Функция, которая отвечает за ввод числа
{
    string number;
    do
    {
        cout << "Введите четырехзначное число с различными цифрами : ";
        cin >> number;
        checkNumber(number);
    } while (checkNumber(number) == false);
    return number;
}

```

```

void bulls_cows(string digit_comp)
{
    int bulls = 0, cows = 0, attemp = 0;
    do {
        bulls = cows = 0;
        //cout << "число, загаданное компьютером " << digit_comp << endl;
        //cout << "число, введенное игроком: " << digit_player << endl;
        std::cout << "Ваш номер хода: " << attemp + 1 << endl;
        string digit_player = vvod_number();
        attemp++;
        for (int i = 0; i < 4; i++)
        {
            if (digit_comp[i] == digit_player[i])
            {

```

```

        bulls++;
    }
}
for (int i = 0; i < 4; i++)
{
    for (int j = 0; j < 4; j++)
    {
        if (digit_player[i] == digit_comp[j]) cows++;
    }
}

std::cout << "Количество коров : " << cows - bulls << endl;
std::cout << "Количество быков: " << bulls << endl;
} while (bulls != 4);

if (bulls == 4) cout << "Поздравляю! Вы отгадали число, задуманное компьютером за
" << attemp;
if (attemp % 10 == 1) cout << " ход";
if (attemp % 10 > 1 && attemp % 10 <= 4) cout << " хода";
if (!(attemp % 10 == 1) && !(attemp % 10 > 1 && attemp % 10 <= 4)) cout << " ходов";
}

int main()
{
    setlocale(LC_ALL, "RUS");
    cout << "                **Добро пожаловать в игру Быки и Коровы!**" << endl;
    cout << "Правила данной игры просты: " << endl;
    cout << "Компьютер генерирует четырехзначное число с различными цифрами, а ваша
задача угадать это число" << endl;
    cout << "При угадывании вам могут встретиться 2 ситуации : быки и коровы" <<
endl;
    cout << "Бык - это цифра, которая есть в загаданном числе и находится на той же
позиции. " << endl;
    cout << "Корова - это цифра, которая так же есть в загаданном числе, но находится не
на своём месте." << endl;
    cout << "Начнем!" << endl << "Начинайте угадывать число, задуманное
компьютером!" << endl << endl << endl;

    string comp_digit = generation();
    bulls_cows(comp_digit);
}

```