Федеральное государственное автономное образовательное учреждение

Санкт-Петербургский политехнический университет Петра Великого

(ФГАОУ ВО «СПБПУ»)

Институт среднего профессионального образования

ОТЧЕТ

Практическая работа №5

О выполнении задачи

по дисциплине: «Основы программирования»

Выполнил студент: Якунин Е.А.

группы 22928/3

Преподаватель: Уцеховский А.А.

Задание #1

Написать программу генерирования последовательности случайных чисел в диапазоне от rand01 до rand02, длина последовательности составляет Nrand. При этом числа в созданной последовательности (или массиве) должны быть уникальны, повторы не допускаются. С учетом этого длина последовательности Nrand не должна превышать количества целых чисел от rand01 до rand02, то есть значения (rand02-rand01+1).

Сформированную последовательность случайных чисел с их номерами (индексами) вывести на консоль.

Выполнить проверку уникальности чисел в последовательности путем попарного сравнения чисел. Так, например, первое число сравнивается со всеми числами последовательности, начиная со второго, второе число сравнивается со всеми числами, начиная с третьего и т.д., предпоследнее число в последовательности сравнивается только с последним числом.

Результаты сравнения выводятся на консоль.

Программу отработать для двух случаев:

rand01 = 1, rand02 = 90, Nrand = 15

rand01 = 1, rand02 = 90, Nrand = 90

Алгоритм решения:

Объявляем переменные и делаем запрос значений у пользователя

Создаём динамический массив

Заполняем массив рандомными числами из диапазона

Создаём цикл для проверки на уникальность

Выводим массив

Код:

#include <iostream>

#include <time.h>

#include <iomanip>

using namespace std;

void check(int rand01, int rand02, int Nrand){

while ((Nrand <= 0) or (Nrand > (rand02 - rand01 + 1)) or (rand01 <= 0) or (rand02 <= 0)){

cout << "Invalid input, please try again" << endl;

cout << "Enter array size => ";

cin >> Nrand;

cout << "Enter smallers random number => ";

cin >> rand01;

cout << "Enter largest random number => ";

cin >> rand02;

}

}

int main(){

srand(time(0));

int rand01=0, rand02=0, Nrand=0;

cout << "Enter array size => ";

cin >> Nrand;

cout << "Enter smallers random number => ";

cin >> rand01;

cout << "Enter largest random number => ";

cin >> rand02;

check(rand01, rand02, Nrand);

int\* mass= new int[Nrand];

cout << "\nSequence of unique random numbers" << endl;

cout << "Range from " << rand01 << " to " << rand02 << ", total " << Nrand << " numbers" << endl;

for (int i = 0; i < Nrand; i++){

mass[i] = rand() % rand02 + rand01;

}

//uniqueness check loop

for (int i = 0; i < Nrand; i++){

for (int j = Nrand; j > i; j--){

if (mass[i] == mass[j] and i != j){

mass[i] = rand() % rand02 + rand01;

j = 90;

i = 0;

}

}

}

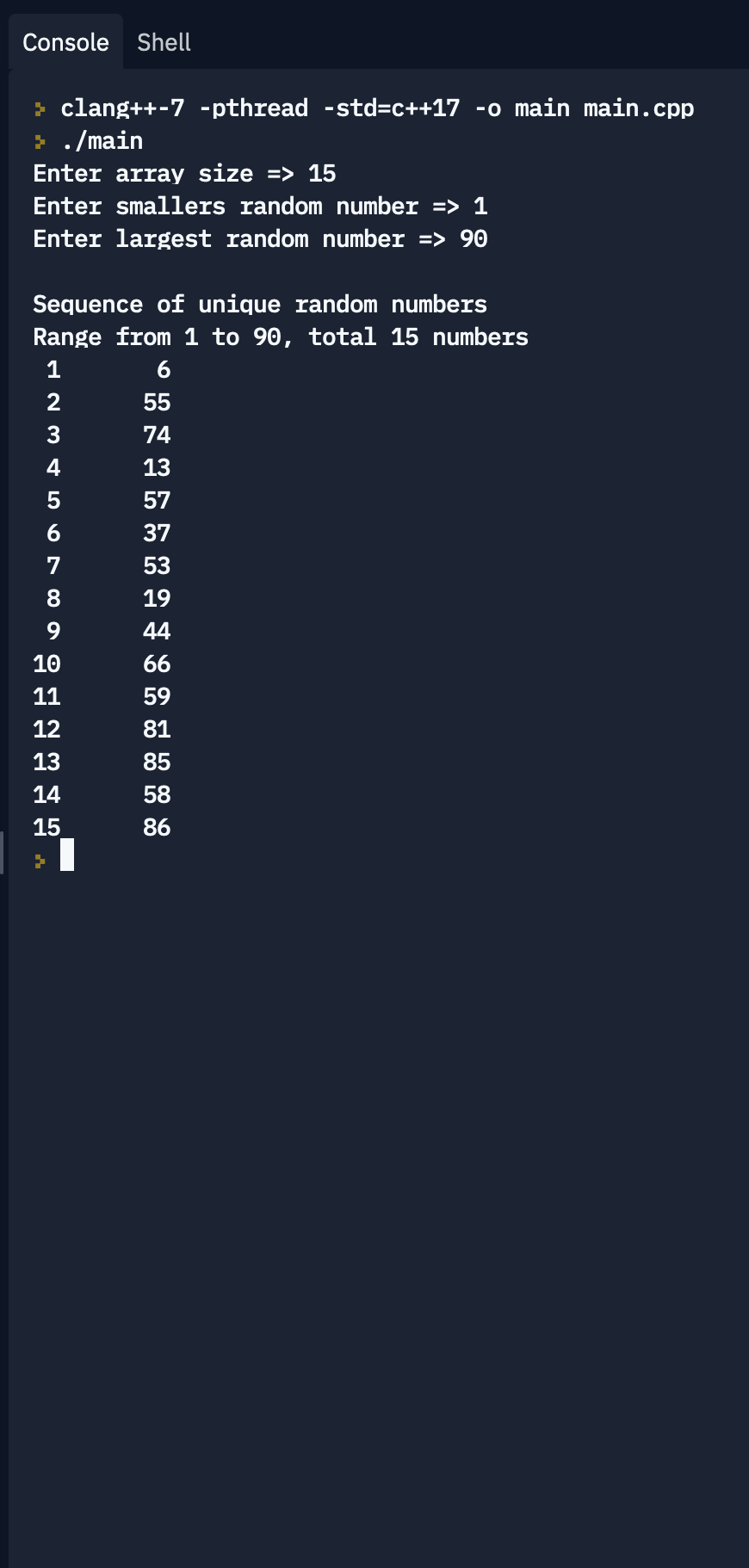
//array output

for (int i = 0; i < Nrand; i++){

cout << setw(2) << i+1 << setw(8) << mass[i] << endl;

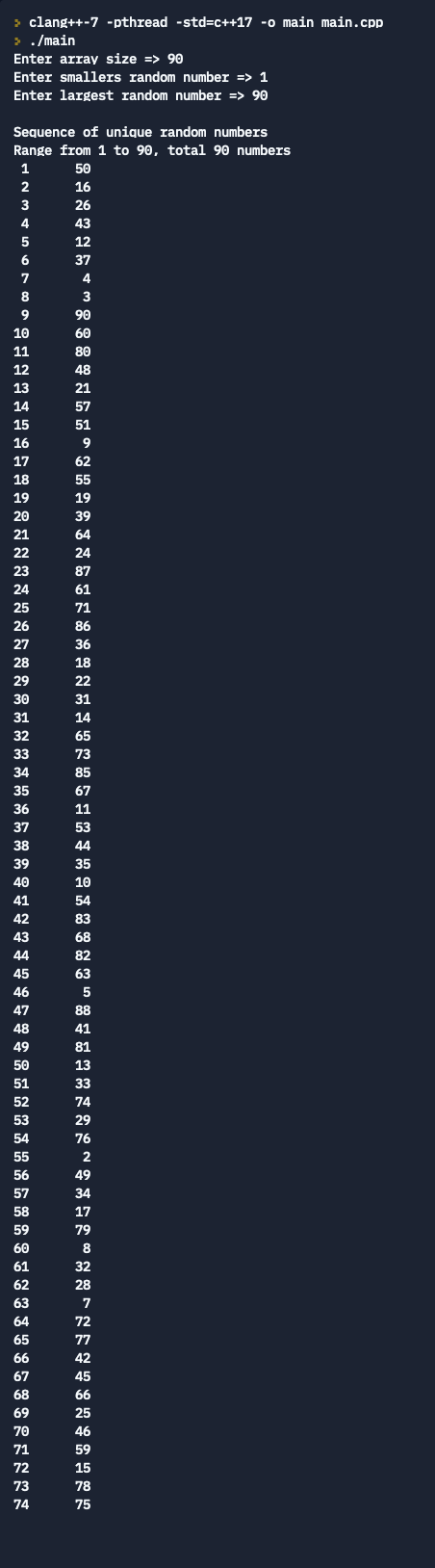
}

}

Результаты выполнения программы:

Nrand = 15:

Nrand = 90:



Задание #2

Написать программу для моделирования игры двух игроков в лото.

Правила игры следующие.

Каждый игрок получает карточку с таблицей в три строки и пять столбцов. В каждой ячейке таблицы содержится случайное число от 1 до 90, Случайные числа не повторяются.

У ведущего в мешочке есть 90 бочонков с цифрами от 1 до 90.

Он случайным образом достает бочонок и объявляет число.

При совпадении числа со значением на карточке одного из игроков, бочонок ставится на эту ячейку в карточке.

Выигрывает тот, кто первым заполнит все ячейки на карточке.

Перед началом игры вывести на консоль сформированные карточки.

Вывести число шагов (бочонков), которое понадобилось для завершения игры, вывести сообщение о том, кто победил и вывести карточки, на которых заполненные ячейки обозначаются нулями.

Провести три игры.

Алгоритм решения:

Объявляем глобальные константы для размера массивов карточек

Создаём функцию для заполнения массива уникальными рандомными числами

Создаём функцию вывода массива карточки

Объявляем массивы и переменные

Создаём цикл для заполнения массива мешочка с бочонками

Создаём цикл для игры, который зависит от счёта игроков

Внутри цикла делаем уникальные рандомные значения бочонков

Выводим результаты игры

Код:

#include <iostream>

#include <iomanip>

using namespace std;

const int rows = 3;

const int cols = 5;

void Array(int mass[rows][cols]){

int rand01 = 1;

int rand02 = 90;

for (int i = 0; i < rows; i++){

for (int j = 0; j < cols; j++){

mass[i][j] = rand() % rand02 + rand01;

}

}

for (int i = 0; i < rows; i++){

for (int k = rows; k > i; k--){

for (int j = 0; j < cols; j++){

for (int l = cols; l > j; l--){

if (mass[i][j] == mass[k][l] and i != k and j != l){

mass[i][j] = rand() % rand02 + rand01;

j = 0;

k = rows;

i = 0;

l = cols;

}

}

}

}

}

}

//Array output

void cout\_array(int mass[rows][cols]){

for (int i = 0; i < rows; i++){

cout << "----------------" << endl;

for (int j = 0; j < cols; j++){

cout << setw(2) << mass[i][j] << setw(1) << '|';

}

cout << endl;

}

cout << "----------------" << endl;

cout << endl;

}

int main(){

srand(time(0));

int list1[rows][cols];

int list2[rows][cols];

int bag[90];

int bochka = 0;

int Count1 = 0;

int Count2 = 0;

int games = 0;

for (int i = 0; i < 90; i++){

bag[i] = i + 1;

}

cout << "Lotto game, 2 players, cards 3Х5, numbers from 1 to 90" << endl;

cout << endl;

Array(list1);

Array(list2);

cout << "Generated cards: " << endl;

cout << endl;

cout << "Player's A card: " << endl;

cout\_array(list1);

cout << endl;

cout << "Player's B card: " << endl;

cout\_array(list2);

//Game loop

while (Count1 != rows \* cols and Count2 != rows \* cols){

int step2 = rand() % 90;

bochka = bag[step2];

bag[step2] = 0;

if (bochka != 0)

games++;

for (int i = 0; i < rows; i++){

for (int j = 0; j < cols; j++){

if (list1[i][j] == bochka and bochka != 0){

list1[i][j] = 0;

Count1++;

}

if (list2[i][j] == bochka and bochka != 0){

list2[i][j] = 0;

Count2++;

}

}

}

}

cout << endl;

cout << "The game has started!" << endl;

cout << endl;

cout << "Came completed on move: " << games << endl;

if (Count1 > Count2)

cout << "Player A wins!" << endl;

else

cout << "Player B wins!" << endl;

cout << endl;

cout << "Player's A card: " << endl;

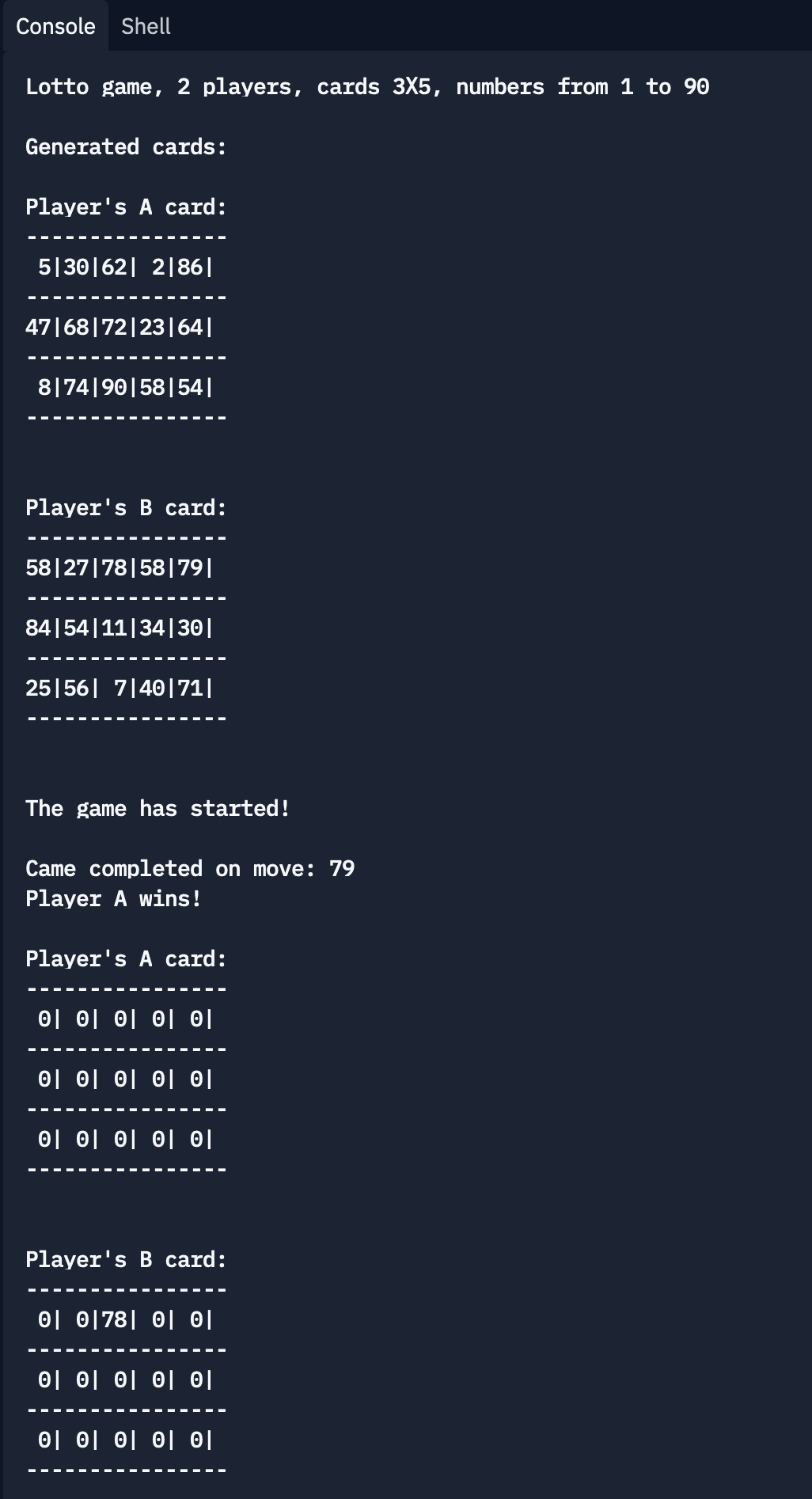
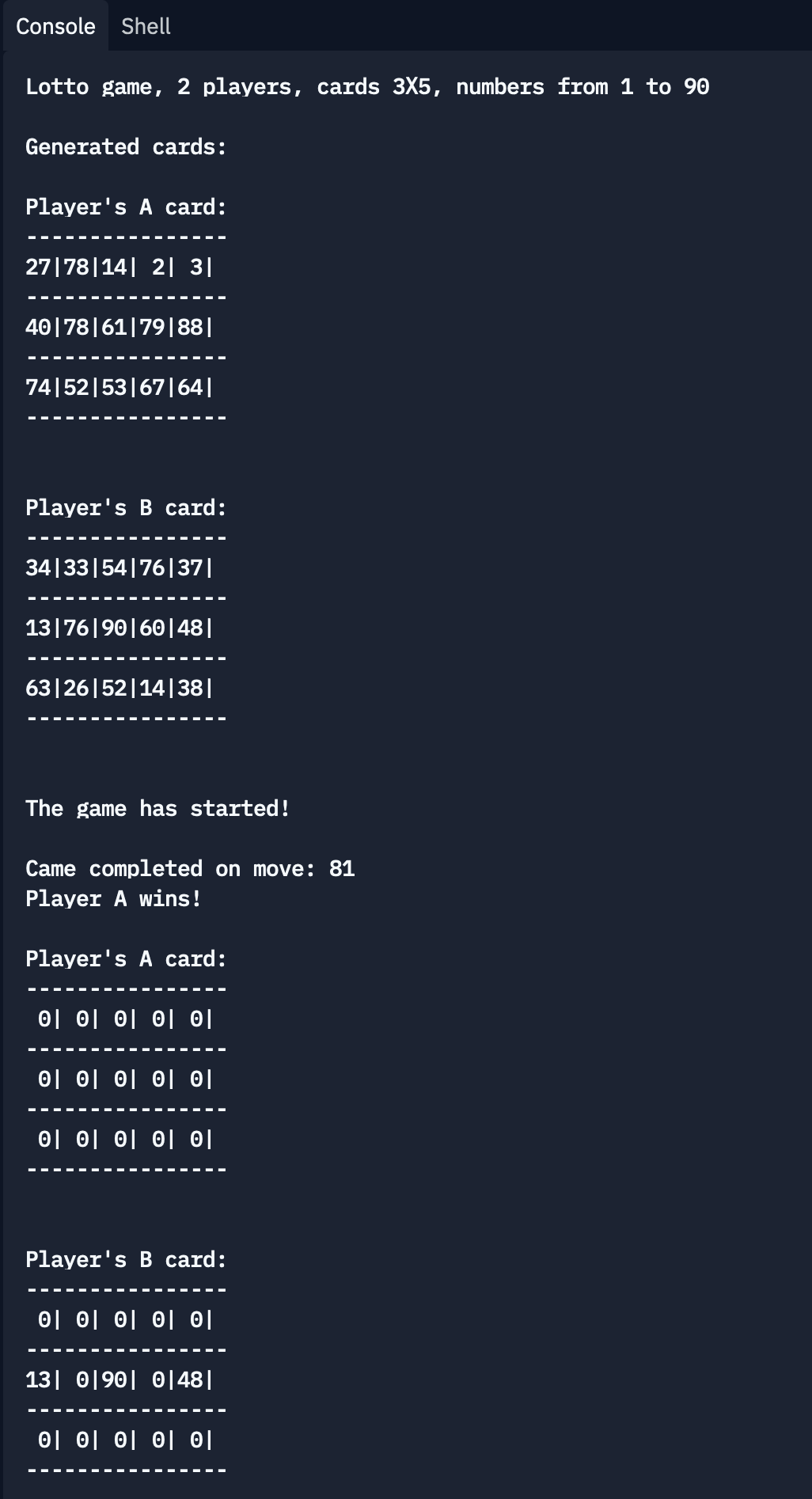
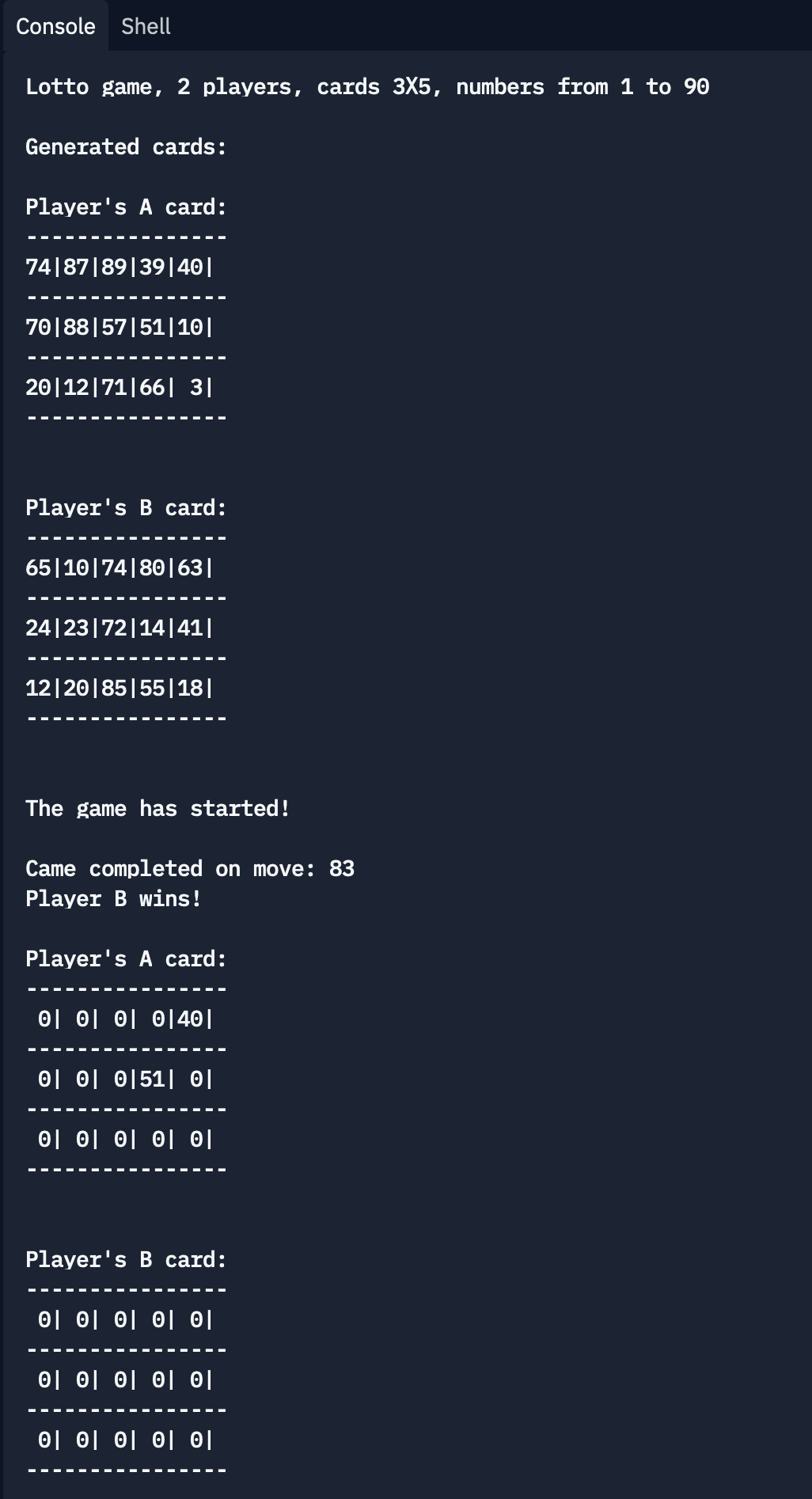
cout\_array(list1);

cout << endl;

cout << "Player's B card: " << endl;

cout\_array(list2);

}

Результаты выполнения программы: