МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«МИРЭА – Российский технологический университет»

РТУ МИРЭА

Институт информационных технологий (ИТ)
Кафедра инструментального и прикладного программного обеспечения (ИиППО)

Дисциплина «Программирование на языке Джава»

ОТЧЕТ
ПО ПРАКТИЧЕСКОМУ ЗАНЯТИЮ №7

Выполнил студент группы ИНБО-02-20 Лукьяненко Д.В.

Принял Степанов П.В.

Практическая работа выполнена «___»_____2021 г.

«_____» «___»_____2021 г.
Отметка о выполнении

Москва – 2021 г.

СОДЕРЖАНИЕ

Цель работы

Цель данной практической работы – изучение на практике приемов работы со стандартными контейнерными классами Java Collection Framework.

Задание

Напишите программу в виде консольного приложения, которая моделирует карточную игру «пьяница» и определяет, кто выигрывает. В игре участвует 10 карт, имеющих значения от 0 до 9, большая карта побеждает меньшую; карта «0» побеждает карту «9».

Карточная игра “ В пьяницу”. В этой игре карточная колода раздается поровну двум игрокам. Далее они открывают по одной верхней карте, и тот, чья карта старше, забирает себе обе открытые карты, которые кладутся под низ его колоды. Тот, кто остается без карт, - проигрывает.

Для простоты будем считать, что все карты различны по номиналу и что самая младшая карта побеждает самую старшую карту (“шестерка берет туз”). Игрок, который забирает себе карты, сначала кладет под низ своей колоды карту первого игрока, затем карту второго игрока (то есть карта второго игрока оказывается внизу колоды).

Входные данные.

Программа получает на вход две строки: первая строка содержит 5 карт первого игрока, вторая - 5 карт второго игрока. Карты перечислены сверху вниз, то есть каждая строка начинается с той карты, которая будет открыта первой.

Выходные данные.

Программа должна определить, кто выигрывает при данной раздаче, и вывести слово first или second, после чего вывести количество ходов, сделанных до выигрыша. Если на протяжении 106 ходов игра не заканчивается, программа должна вывести слово botva.

Пример ввода.

1 3 5 7 9
2 4 6 8 0
second 5

Репозиторий

Ссылка: https://github.com/neluckoff/mirea-java-lessons/tree/master/src/ru/luckoff/mirea/practice_7

Выполнение работы

Первым делом мною был создан класс Main, в котором я создал 2 стека, для первого игрока и для его противника. Для того, чтобы не заполнять стеки самому, я сделал небольшую функцию заполнения стека рандомными числами от 0 до 9.

Далее я реализовал сам “игровой процесс” и красиво оформил вывод программы.

Ниже мною будет представлена UML Диаграмма, которая была создана после написании программы (рис. 1).

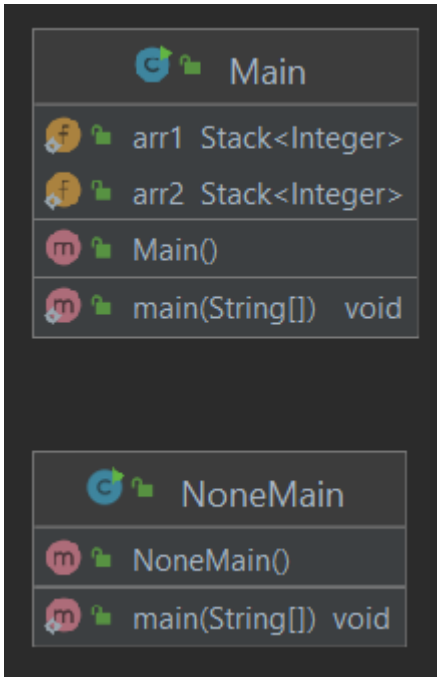

Рисунок 1 – Полученная UML Диаграмма

Код выполненной работы

Здесь в нескольких скриншотах можно увидеть как выглядит код выполненного задания и результат его работы.

```
public static Stack<Integer> arr1;
public static Stack<Integer> arr2;

public static void main(String[] args) {
    arr1 = new Stack<>();
    arr2 = new Stack<>();
    int[] a1 = new int[5];
    int[] a2 = new int[5];

    for (int i = 0; i < 5; i++) {
        a1[i] = ((int) (Math.random() * 9));
        a2[i] = ((int) (Math.random() * 9));
        arr2.push(a1[i]);
        arr1.push(a2[i]);
    }

    System.out.println(arr1);
    System.out.println(arr2);
```

Рисунок 2 – Создание, заполнение и вывод стеков

```
int maxSteps = 106;
int i;
for (i = 0; !arr1.isEmpty() && !arr2.isEmpty() && i < maxSteps; i++) {
    int first = arr1.pop();
    int second = arr2.pop();

    if ((first == 0 && second == 9) || first > second && (first != 9 && second != 0)) {
        arr1.push(first);
        arr1.push(second);
    } else if (first < second || (first == 9 && second == 0)) {
        arr2.push(first);
        arr2.push(second);
    }
}
```

Рисунок 2 – Реализация игры

```
if(maxSteps == i) {
    System.out.println("botva");
} else if (arr1.isEmpty()) {
    System.out.println("second " + i);
} else if (arr2.isEmpty()) {
        System.out.println("first " + i);
}
```

Рисунок 3 – Вывод результата игры

Вывод

В результате выполнения данной практической работы я изучил на практике приемы работы со стандартными контейнерными классами Java Collection Framework.