Министерство науки и высшего образования РФ

ФГАОУ ВО Пермский национальный исследовательский

политехнический университет

Кафедра «Вычислительная математика, механика и биомеханика»

Отчет по лабораторной работе № 2

тема «Игра + массив»

по дисциплине «Учебно-исследовательская работа»

Выполнил: студент группы ИСТ-22-1б Богданов Д.О.

Проверила Сахабутдинова Л.Р.

Пермь, 2023

**Содержание**

[Задание 1. 3](#_Toc149422484)

[1.1. Постановка задачи 3](#_Toc149422485)

[1.2. Решение задачи, код программы 3](#_Toc149422486)

[1.3. Ход решения 4](#_Toc149422487)

[1.4. Тестирование работы программы с проверкой 4](#_Toc149422488)

[Задание 2. 5](#_Toc149422489)

[2.1. Постановка задачи 5](#_Toc149422490)

[2.2. Решение задачи, код программы 5](#_Toc149422491)

[2.3. Ход решения 6](#_Toc149422492)

[2.4. Тестирование работы программы с проверкой 6](#_Toc149422493)

[Задание 3. 7](#_Toc149422494)

[3.1. Постановка задачи 7](#_Toc149422495)

[3.2. Решение задачи, код программы 7](#_Toc149422496)

[3.3. Ход решения 9](#_Toc149422497)

[3.4. Тестирование работы программы с проверкой 9](#_Toc149422498)

[Задание 4. 10](#_Toc149422499)

[4.1. Постановка задачи 10](#_Toc149422500)

[4.2. Решение задачи, код программы 10](#_Toc149422501)

[4.3. Ход решения 11](#_Toc149422502)

[4.4. Тестирование работы программы с проверкой 11](#_Toc149422503)

Задание 1.

1.1. Постановка задачи

Создать игру камень-ножницы бумага.

1.2. Решение задачи, код программы

import random  
ver = 0  
while ver == 0:  
 player = int(input("1 - камень; 2 - ножницы; 3 - бумага\n"))  
 if player == 1 or player == 2 or player == 3:  
 ver = 1  
 else:  
 while player not in [1, 2, 3]:  
 print("Неверное значение, попробуйте еще раз")  
 player = int(input("1 - Камень; 2 - Ножницы; 3 - Бумага\n"))  
 if player == 1 or player == 2 or player == 3:  
 ver = 1  
 choices = {1: "Камень", 2: "Ножницы", 3: "Бумага"}  
 comp = random.randint(1, 3)  
  
 print("(Вы) - ", choices[player])  
 print("(ИИ) - ", choices[comp])  
  
 if player == comp:  
 win = 0  
 elif (player == 1 and comp == 2) or (player == 2 and comp == 3) or (player == 3 and comp == 1):  
 win = 1  
 elif (player == 1 and comp == 3) or (player == 2 and comp == 1) or (player == 3 and comp == 2):  
 win = 2  
  
 if win == 0:  
 print("НИЧЬЯ")  
 elif win == 1:  
 print("ПОБЕДА ЧЕЛОВЕЧЕСТВА")  
 else:  
 print("ПОБЕДА БЕЗДУШНОЙ МАШИНЫ")

## 1.3. Ход решения

1. Задаем переключатель ver
2. Пока он равен 0 выполняется игра
3. Польз. выбирает чем будет играть
4. Если выбрано значение не из допустимых, заново
5. Компьютер выбирает рандомное значение от 1 до 3
6. При помощи словаря выводим выбор каждого игрока
7. Сравниваем выбор и выбираем победителя

## 1.4. Тестирование работы программы с проверкой

Тестирование работы программы и проверка результатов решения

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Python | |
| 1 |  |  |

Задание 2.

2.1. Постановка задачи

С продолжением игры

2.2. Решение задачи, код программы

import random  
ver = 0  
win = 0  
while ver == 0:  
 player = int(input("1 - Камень; 2 - Ножницы; 3 - Бумага\n"))  
 if player == 1 or player == 2 or player == 3:  
 ver = 1  
 else:  
 while player not in [1, 2, 3]:  
 print("Неверное значение, попробуйте еще раз")  
 player = int(input("1 - Камень; 2 - Ножницы; 3 - Бумага\n"))  
 if player == 1 or player == 2 or player == 3:  
 ver = 1  
 choices = {1: "Камень", 2: "Ножницы", 3: "Бумага"}  
 comp = random.randint(1, 3)  
  
 print("(Вы) - ", choices[player])  
 print("(ИИ) - ", choices[comp])  
  
 if player == comp:  
 win = 0  
 elif (player == 1 and comp == 2) or (player == 2 and comp == 3) or (player == 3 and comp == 1):  
 win = 1  
 elif (player == 1 and comp == 3) or (player == 2 and comp == 1) or (player == 3 and comp == 2):  
 win = 2  
  
 if win == 0:  
 print("НИЧЬЯ")  
 elif win == 1:  
 print("ПОБЕДА ЧЕЛОВЕЧЕСТВА")  
 else:  
 print("ПОБЕДА БЕЗДУШНОЙ МАШИНЫ")  
 cont = int(input("Для продолжения игры нажмите '1'. В ином случае игра завершится.\n"))  
 if cont == 1:  
 ver = 0

## 2.3. Ход решения

1. Ход программы, как и в 1
2. Добавляем проверку на продолжение cont с помощью переключателя ver

## 2.4. Тестирование работы программы с проверкой

Тестирование работы программы и проверка результатов решения

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Python | |
| 1 |  |  |

Задание 3.

3.1. Постановка задачи

Добавить выбор типа игры: турнир или обычная.

3.2. Решение задачи, код программы

import random  
  
ver = 0  
win = 0  
player = 0  
game = 0  
  
  
def check():  
 global ver  
 global player  
 player = int(input("1 - Камень; 2 - Ножницы; 3 - Бумага\n"))  
  
 if player == 1 or player == 2 or player == 3:  
 ver = 1  
 else:  
 while player not in [1, 2, 3]:  
 print("Неверное значение, попробуйте еще раз")  
 player = int(input("1 - Камень; 2 - Ножницы; 3 - Бумага\n"))  
 if player == 1 or player == 2 or player == 3:  
 ver = 1  
  
  
def play\_round(player):  
 global win  
  
 choices = {1: "Камень", 2: "Ножницы", 3: "Бумага"}  
 comp = random.randint(1, 3)  
  
 print("(Вы) - ", choices[player])  
 print("(ИИ) - ", choices[comp])  
  
 if player == comp:  
 win = 0  
 elif (player == 1 and comp == 2) or (player == 2 and comp == 3) or (player == 3 and comp == 1):  
 win = 1  
 elif (player == 1 and comp == 3) or (player == 2 and comp == 1) or (player == 3 and comp == 2):  
 win = 2  
  
 if win == 0:  
 print("НИЧЬЯ")  
 elif win == 1:  
 print("ПОБЕДА ЧЕЛОВЕЧЕСТВА")  
 else:  
 print("ПОБЕДА БЕЗДУШНОЙ МАШИНЫ")  
 return win  
  
  
while game == 0:  
 variant = int(input("Выберите тип игры: 1 - Турнир; 2 - Обычная.\n"))  
 if variant == 1:  
 pc = 0  
 human = 0  
 game = 1  
 count = int(input("Выберите кол-во раундов \n"))  
 while count >= 1:  
 check()  
 win1 = play\_round(player)  
 count -= 1  
 ver = 0  
 if win == 1:  
 human += 1  
 if win == 2:  
 pc += 1  
 if human > pc:  
 print(f"Вы победили со счетом {human} : {pc}")  
 if human < pc:  
 print(f"ИИ победил со счетом {pc} : {human}")  
 if human == pc:  
 print(f"Ничья со счетом {pc} : {human}")  
  
 if variant == 2:  
 while ver == 0:  
 check()  
 win2 = play\_round(player)  
 game = 1  
 ver = 0  
  
 cont = int(input("Для продолжения игры нажмите '1'. В ином случае игра завершится.\n"))  
 if cont == 1:  
 game = 0

## 3.3. Ход решения

1. Задаем новый переключатель game, старый ver, обозначаем player и win для задания функций
2. На основе кода выбора игрока с проверкой и кода раунда из предыдущих 2 заданий создаем функции check и play\_round
3. Пока game = 0, идет цикл на выбор игры.
4. При выборе турнира (1) задаем переменные-счетчики pc и human для подсчета конечного итога, задаем выбор кол-ва раундов count. Пока count >= 1 игра идет, совершается подсчет по кол-ву win у компа и игрока. Итог: определяется победитель или ничья, выводится счет
5. При выборе обычной игры (2) все так же, как и во 2 задании, но без продолжения свободной игры.
6. Вводим проверку для продолжения с помощью счетчика game.

## 3.4. Тестирование работы программы с проверкой

Тестирование работы программы и проверка результатов решения

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Python | |
| 1 |  |  |

Задание 4.

4.1. Постановка задачи

Необходимо заполнить матрицу А[n\*n] по одному из описанных алгоритов.

Затем отразить ее "зеркально" вправо, получив при этом матрицу размера B[n\*2n].

Полученную матрицу B отразить "зеркально" вниз, полив матрицу C [2n\*2n]

4.2. Решение задачи, код программы

n = int(input("Введите размер матрицы A\n"))  
A = []  
number = 1  
for i in range(n):  
 row = []  
 for j in range(n):  
 if i % 2 == 0:  
 row.append(number)  
 number += 1  
 else:  
 row.insert(0, number)  
 number += 1  
 A.append(row)  
  
for i in range(n):  
 for j in range(n):  
 print(A[i][j], end=", ")  
 print()  
print()  
B = []  
for row in A:  
 B.append(row + row[::-1])  
for row in B:  
 print(row)  
print()  
C = B + B[::-1]  
for row in C:  
 print(row)

## 4.3. Ход решения

1. Создаем пустой список-матрицу А
2. Задаем переменную number, числа, что буду идти змейкой
3. Заполняем строки от 0 до n-1. Если число по модулю 2 = 0, то идем слева-направо, если нет, то справа-налево. После каждого заполненной строки добавляем ее в матрицу А
4. Выводим матрицу поэлементно ради разнообразия.
5. Создаем пустой список -матрицу B
6. Заполняем B строками из суммы строк А и их зеркальных отражений через slicing
7. Создаем C состоящую из B + ее зеркального отражения

## 4.4. Тестирование работы программы с проверкой

Тестирование работы программы и проверка результатов решения

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Python | |
| 1 |  |  |