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 группу ИСТ-22-1б Зверев А.А.

Проверил: Сахабутдинова Ляйсан Рамилевна

Пермь, 2023

**Содержание**

[Задание 1 3](#_Toc149516379)

[1.1. Постановка задачи 3](#_Toc149516380)

[1.2. Код программы 3](#_Toc149516381)

[1.3. Тестирование работы программы 4](#_Toc149516382)

[Задание 2 4](#_Toc149516383)

[2.1. Постановка задачи 4](#_Toc149516384)

[2.2. Код программы 4](#_Toc149516385)

[2.3. Тестирование работы программы 6](#_Toc149516386)

[Задание 3 6](#_Toc149516387)

[3.1. Постановка задачи 6](#_Toc149516388)

[3.2. Код программы 6](#_Toc149516389)

[3.3. Тестирование работы программы 9](#_Toc149516390)

[Задание 4 11](#_Toc149516391)

[4.1. Постановка задачи 11](#_Toc149516392)

[4.2. Код программы 11](#_Toc149516393)

[4.3. Тестирование работы программы 12](#_Toc149516394)

# Задание 1

## 1.1. Постановка задачи

Реализовать просто разовую игру. Для запуска раунда надо перезапускать скрипт.

## 1.2. Код программы

|  |  |
| --- | --- |
| def whowins(player1, player2):  if (player1 == 1 and player2 == 3):  print("Компьютер выйграл ((")  elif (player1 == 2 and player2 == 1):  print("Компьютер тебя победил ((")  elif (player1 == 3 and player2 == 2):  print("Востание машин, тебя сильно оскорбила машина и да, ты проиграл")  elif (player1 == player2):  print("Ничья")  else: print("Вау, да ты молодец, ты победил") | Функция для проверки кто выйграл |
| def cheker():  flag = 0;  while (flag == 0):  player\_input = input("Введите число от 1 до 3: ")  if (player\_input == '1' or player\_input == '2' or player\_input == '3'):  flag = 1  return player\_input | Функция для проверки ввел ли пользователь 1, 2 или 3, возвращает что ввел пользователь |
| def whatsim(inp):  if inp == 1: s = 'камень'  elif inp == 2: s = 'ножницы'  else: s = 'бумага'  return s | Функция, возвращающая какой из ходов выбран |
| print("КАМЕНЬ-НОЖНИЦЫ-БУМАГА")  print("Правила:"  " 1 - камень, 2 - ножницы, 3 - бумага")  print("Камень бьет ножницы, ножницы бьют бумагу, бумага бьет камень")  player\_input = int(cheker())  pc\_input = random.randint(1,3)  print("Комппьютер выбрал ", pc\_input, whatsim(pc\_input))  whowins(player\_input,pc\_input) | Вывод правил, пользователь выбирает, компьютер выбирает, сравнивает и выбирает победителя |

## 1.3. Тестирование работы программы

|  |  |
| --- | --- |
| № проверки | Вид |
| 1 |  |
| 2 |  |
| 3 |  |

# Задание 2

## 2.1. Постановка задачи

Реализовать игру с последующим приглашением повторить игру.

## 2.2. Код программы

|  |  |
| --- | --- |
| def whowins(player1, player2):  if (player1 == 1 and player2 == 3):  print("Компьютер выйграл ((")  elif (player1 == 2 and player2 == 1):  print("Компьютер тебя победил ((")  elif (player1 == 3 and player2 == 2):  print("Востание машин, тебя сильно оскорбила машина и да, ты проиграл")  elif (player1 == player2):  print("Ничья")  else: print("Вау, да ты молодец, ты победил") | Функция для проверки кто выйграл |
| def cheker(k):  flag = 0;  if k == 1:  while (flag == 0):  player\_input = input("Введите число от 1 до 3: ")  if (player\_input == '1' or player\_input == '2' or player\_input == '3'):  flag = 1  else:  while (flag == 0):  player\_input = input("Введите 1 для продолжения, 0 для выхода: ")  if (player\_input == '1' or player\_input == '0'):  flag = 1  return player\_input | Функция для проверки ввел ли пользователь 1, 2 или 3, возвращает что ввел пользователь, так же для выхода из программы 1 или 0 |
| def whatsim(inp):  if inp == 1: s = 'камень'  elif inp == 2: s = 'ножницы'  else: s = 'бумага'  return s | Функция, возвращающая какой из ходов выбран |
| flag = 1  while (flag == 1):  print("КАМЕНЬ-НОЖНИЦЫ-БУМАГА")  print("Правила:"  " 1 - камень, 2 - ножницы, 3 - бумага")  print("Камень бьет ножницы, ножницы бьют бумагу, бумага бьет камень")  player\_input = int(cheker(1))  pc\_input = random.randint(1, 3)  print("Комппьютер выбрал ", pc\_input, " - ", whatsim(pc\_input))  whowins(player\_input, pc\_input)  flag = int(cheker(2)) | Вывод правил, пользователь выбирает, компьютер выбирает, сравнивает и выбирает победителя, будет работать до того момента, пока не пользователь при вопросе повторной игры не выберет 0 |

## 2.3. Тестирование работы программы

|  |  |
| --- | --- |
| № проверки | Виды |
| 1 |  |
| 2 |  |

# Задание 3

## 3.1. Постановка задачи

Реализовать турнирную игру. При запуске игры пользователю выводится сообщение с выбором режима игры. Если выбран просто игра, то возвращается режим из ур.2. Если выбирает турнир, то предлагается ввести кол-во раундов. После этого проигрывается заданное кол-во раундов и выводится итог турнира. Например: «Поздравляю, Вы победили со счетом 2:1»

## 3.2. Код программы

|  |  |
| --- | --- |
| def whowins(player1, player2):  if (player1 == 1 and player2 == 3):  out = "Компьютер выйграл (("  elif (player1 == 2 and player2 == 1):  out = "Компьютер выйграл (("  elif (player1 == 3 and player2 == 2):  out = "Компьютер выйграл (("  elif (player1 == player2):  out = "Ничья"  else:  out = "Вау, да ты молодец, ты победил"  return out | Функция для проверки кто выйграл |
| def cheker(k):  flag = 0;  if k == 1:  while (flag == 0):  player\_input = input("Введите число от 1 до 3: ")  if (player\_input == '1' or player\_input == '2' or player\_input == '3'):  flag = 1  else:  while (flag == 0):  player\_input = input("Введите 1 для продолжения, 0 для выхода: ")  if (player\_input == '1' or player\_input == '0'):  flag = 1  return player\_input | Функция для проверки ввел ли пользователь 1, 2 или 3, возвращает что ввел пользователь, так же для выхода из программы 1 или 0 |
| def whatsim(inp):  if inp == 1: s = 'камень'  elif inp == 2: s = 'ножницы'  else: s = 'бумага'  return s | Функция, возвращающая какой из ходов выбран |
| def standart():  flag = 1  while (flag == 1):  print("КАМЕНЬ-НОЖНИЦЫ-БУМАГА")  print("Правила:"  " 1 - камень, 2 - ножницы, 3 - бумага")  print("Камень бьет ножницы, ножницы бьют бумагу, бумага бьет камень")  player\_input = int(cheker(1))  pc\_input = random.randint(1, 3)  print("Комппьютер выбрал ", pc\_input, " - ", whatsim(pc\_input))  whowins(player\_input, pc\_input)  flag = int(cheker(2)) | Стандартная игра, как в задании 2 |
| def tournir(i):  player\_wins = 0  pc\_wins = 0  draw = 0  while (i != 0):  print("КАМЕНЬ-НОЖНИЦЫ-БУМАГА")  print("Правила:"  " 1 - камень, 2 - ножницы, 3 - бумага")  print("Камень бьет ножницы, ножницы бьют бумагу, бумага бьет камень")  player\_input = int(cheker(1))  pc\_input = random.randint(1, 3)  print("Комппьютер выбрал ", pc\_input, " - ", whatsim(pc\_input))  i -= 1  s = whowins(player\_input,pc\_input)  if (s == "Компьютер выйграл (("): pc\_wins += 1  elif (s == "Ничья"): draw += 1  else: player\_wins += 1  print("Ты выйграл ", player\_wins, "матчей"  "Компьютер выйиграл", pc\_wins, "матчей",  "Ничья ", draw)  if (player\_wins == pc\_wins):  print("Итог ничья, играй еще")  standart()  elif (player\_wins > pc\_wins): print("Ты выйграл турнир")  else: print("Компьютер восстал") | Турнирная игра с конечным выводом выйгранных игр с обоих сторон |
| print("Турнир - 1 или стандарт - 2, выход 0")  flag = input()  if flag.isdigit():  flag = int(flag)  while (flag != 1 and flag != 2 and flag != 0):  print("Турнир - 1 или стандарт -2, выход 0")  flag = input()  if flag.isdigit():  flag = int(flag)  if flag == 1:  rounds = int(input("Сколько раундов: "))  tournir(rounds)  elif flag == 2:  standart() | Выбор игроком вида игры и ее запуск |

## 3.3. Тестирование работы программы

|  |  |
| --- | --- |
| № проверки | Вид |
| 1 |  |
| 2 |  |

# 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. Код программы

|  |  |
| --- | --- |
| def zmeika\_nxn(n):  arr1 = [0] \* n  for i in range(n):  arr1[i] = [0] \* n  for i in range(n):  for j in range(n):  if (i % 2 == 0):  arr1[i][j] = n \* i + j + 1  else:  arr1[i][j] = n \* (i + 1) - j  return arr1 | Заполнение матрицы змейкой размерности n на n |
| def zmeika\_nx2n(n, arr1):  arr2 = [0] \* n  for i in range(n):  arr2[i] = [0] \* n \* 2  for i in range(n):  for j in range(n):  arr2[i][j] = arr1[i][j]  for i in range(n):  for j in range(n \* 2):  arr2[i][n \* 2 - j - 1] = arr2[i][j]  return arr2 | Отражение матрицы змейка “зеркально” вправо, тем самым получив матрицу размерности n на 2n |
| def zmeika\_2nx2n(n, arr2):  arr3 = [0] \* n \* 2  for i in range(n \* 2):  arr3[i] = [0] \* n \* 2  for i in range(n):  for j in range(n \* 2):  arr3[i][j] = arr2[i][j]  for i in range(n \* 2):  for j in range(n \* 2):  arr3[n \* 2 - i - 1][j] = arr3[i][j]  return arr3 | Матрица n на 2n отраженная “зеркально” вниз, тем самым получив матрицу размерности 2n на 2n |
| flag = 0  while flag != 1:  n = input('Input length of double array: ')  if (n.isdigit()):  n = int(n)  flag = 1  else:  print("Incorrect input")  flag = 0  arrA = zmeika\_nxn(n)  print("Array A= ")  for i in range(n):  for j in range(n):  print('%2.f' % arrA[i][j], end=' ')  print()  arrB = zmeika\_nx2n(n, arrA)  print("Array B= ")  for i in range(n):  for j in range(n\*2):  print('%2.f' % arrB[i][j], end=' ')  print()  arrC = zmeika\_2nx2n(n, arrB)  print("Array B= ")  for i in range(n\*2):  for j in range(n\*2):  print('%2.f' % arrC[i][j], end=' ')  print() | Спрашиваем у пользователя какой размерности должна быть матрица A, после программа выводит матрицу A(n x n), B(n x 2n), C(2n x 2n). |

## 4.3. Тестирование работы программы

|  |  |
| --- | --- |
| № проверки | Вид |
| 1 |  |
| 2 |  |