|  |  |  |
| --- | --- | --- |
|  |  |  |
| МИНОБРНАУКИ РОССИИ | | |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |
| **Институт комплексной безопасности и специального приборостроения** | | |
|  |  |  |

Кафедра КБ-4 «Управление и моделирование систем»

**Отчет**

По проекту:  
Виртуальное Казино на Python

Выполнил:

Студент 3 курса

Группа ББСО-02-19

Шифр 19Б1509

Доронин Владислав Алексеевич

Москва, 2021

Мой проект называется “Виртуальное Казино”. В нём я реализую 5 консольных игр, 3 из которых можно сыграть против компьютера, а 2 других с человеком.

Опишу по пунктам принцип и логику работы каждой игры.  
1. Меню

С помощью меню можно выбирать игру, а также выходить из программы.

while True:  
 print("1. Играть в 21")  
 print("2. Играть в рулетку")  
 print("3. Играть в Однорукого Бандита")  
 print("4. Играть в Крестики-Нолики на двоих")  
 print("5. Играть в Камень-Ножницы-Бумага на двоих")  
 print("6. Выход")  
 ch = input("Выберите пункт: ")

2. Игра 21

Самая объемная по количеству строк игра, которая была реализована с помощью оптимальной стратегии. Сначала карты набирает игрок, потом компьютер.   
Игрок:

for mast, numb in player\_cards:  
 if numb == 'A':  
 if player\_sum + 11 > 21:  
 player\_sum += 1  
 else:  
 player\_sum += 11  
 else:  
 player\_sum += detect[numb]  
  
computer\_cards.append(cards.pop())  
computer\_cards.append(cards.pop())  
for mast, numb in computer\_cards:  
 if numb == 'A':  
 if computer\_sum + 11 > 21:  
 computer\_sum += 1  
 else:  
 computer\_sum += 11  
 else:  
 computer\_sum += detect[numb]

Компьютер:

while True:  
 if computer\_sum + 5 <= 21:  
 card = cards.pop()  
 numb = card[1]  
 computer\_cards.append(card)  
  
 if numb == 'A':  
 if computer\_sum + 11 > 21:  
 computer\_sum += 1  
 else:  
 computer\_sum += 11  
 else:  
 computer\_sum += detect[numb]  
 else:  
 break

3. Игра Рулетка  
Реализована по тому же принципу, как и в реальном казино. Шанс победы компьютера немного выше из-за клетки Зеро.

def roll(self):  
 num = randint(0, 36)  
  
 isdigit = False  
 player = input('Ставим на Красное или Черное? (red/black)')  
 if player.isdigit():  
 player = int(player)  
 isdigit = True  
 elif player not in ['red', 'black']:  
 return None  
  
 red = False  
 black = False  
  
 if num == 0:  
 print('Зеро')  
 elif num in self.reds:  
 print(num, 'Красное')  
 red = True  
 elif num in self.blacks:  
 print(num, 'Черное')  
 black = True

4. Игра Однорукий Бандит

Есть 5 различных символов, которые могут выпасть. Работает по логике:   
Если хотя бы 2 одинаковых символа составляют линию, то это победа. Если выпадает 3 одинаковых символа – это Джекпот. (в качестве символов были использованы эмоджи, которые не отображаются в Word)

s1 = random.SystemRandom().choice(['🍒', '🍌', '💲', '7', '🔔'])  
s2 = random.SystemRandom().choice(['🍒', '🍌', '💲', '7', '🔔'])  
s3 = random.SystemRandom().choice(['🍒', '🍌', '💲', '7', '🔔'])  
  
print ('| - ' + s1 + ' - | - ' + s2 + ' - | - ' + s3 + ' - |')  
time.sleep(2)  
if s1 == s2 or s2 == s3:  
 print ('Вы победили!')  
elif s1 == s2 == s3:  
 print ('Вы сорвали Джекпот!')  
else:  
 print ('Вы проиграли!')

5. Игра Крестики-Нолики на двоих

Реализовано поле в консоли, с помощью которого можно видеть расположение символов.

board = list(range(1, 10))  
  
def draw\_board(board):  
 print("-" \* 13)  
 for i in range(3):  
 print("|", board[0 + i \* 3], "|", board[1 + i \* 3], "|", board[2 + i \* 3], "|")  
 print("-" \* 13)

def take\_input(pl\_ch):  
 valid = False  
 while not valid:  
 pl\_ans = input("Куда поставим " + pl\_ch + "? ")  
 try:  
 pl\_ans = int(pl\_ans)  
 except:  
 print("Некорректный ввод. Вы уверены, что ввели число?")  
 continue  
 if pl\_ans >= 1 and pl\_ans <= 9:  
 if (str(board[pl\_ans - 1]) not in "XO"):  
 board[pl\_ans - 1] = pl\_ch  
 valid = True  
 else:  
 print("Клетка занята!")  
 else:  
 print("Некорректный ввод. Введите число от 1 до 9.")

def check\_win(board):  
 win\_coord = ((0, 1, 2), (3, 4, 5), (6, 7, 8), (0, 3, 6), (1, 4, 7), (2, 5, 8), (0, 4, 8), (2, 4, 6))  
 for each in win\_coord:  
 if board[each[0]] == board[each[1]] == board[each[2]]:  
 return board[each[0]]  
 return False  
  
def main(board):  
 counter = 0  
 win = False  
 while not win:  
 draw\_board(board)  
 if counter % 2 == 0:  
 take\_input("X")  
 else:  
 take\_input("O")  
 counter += 1  
 if counter > 4:  
 tmp = check\_win(board)  
 if tmp:  
 print(tmp, "выиграл!")  
 break  
 if counter == 9:  
 print("Ничья!")  
 break  
 draw\_board(board)  
  
main(board)

6. Игра Камень-Ножницы-Бумага на двоих

Очень простая реализация. Решил сделать выбор предмета через однозначную цифру, т.к у названий предметов разная длина, из-за чего можно интуитивно понять, что ввёл соперник.

p1 = input("Игрок 1. Камень (1), Ножницы (2), Бумага (3) ?\n")  
p1 = p1.lower()  
  
print()  
  
p2 = input("Игрок 2. Камень (1), Ножницы (2), Бумага (3) ?\n")  
p2 = p2.lower()  
  
print()  
  
if (p1 == "1"):  
 if (p2 == "1"):  
 print("Ничья!")  
 elif (p2 == "3"):  
 print("Игрок 2 победил!")  
 elif (p2 == "2"):  
 print("Игрок 1 победил!")  
elif (p1 == "3"):  
 if (p2 == "1"):  
 print("Игрок 1 победил!")  
 elif (p2 == "3"):  
 print("Ничья!")  
 elif (p2 == "2"):  
 print("Игрок 2 победил!")  
elif (p1 == "2"):  
 if (p2 == "1"):  
 print("Игрок 2 победил!")  
 elif (p2 == "3"):  
 print("Игрок 1 победил!")  
 elif (p2 == "2"):  
 print("Ничья!")  
else:  
 print("Неправильный ввод. Попробуйте еще раз.")