**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ,**

**СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТЕЛЕКОММУНИКАЦИЙ ИМ. ПРОФ. М.А. БОНЧ-БРУЕВИЧА» (СПбГУТ)**

Факультет: Инфокоммуникационных сетей и систем Кафедра: Защищенных систем связи

Дисциплина: Защищенные операционные системы

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ

Реализация игры «Крестики-нолики»

*(тема отчета)*

Направление/специальность подготовки

10.03.01 Информационная безопасность

*(код и наименование направления/специальности)*

Студент:

Емельянов Н.Р. ИКБ-32

*(Ф.И.О., № группы) (подпись)*

Михайлов И.Д. ИКБ-32

*(Ф.И.О., № группы) (подпись)*

Преподаватель:

Асс. Смирнов Д.Н.

*(Должность, Ф.И.О. преподавателя) (подпись)*

Санкт-Петербург 2024

**Цель**

Реализовать игру «Крестики-нолики».

**Задачи**

1. Создать функцию для отображения игрового поля;
2. Реализовать механизм для хода игроков (заполнение клетки "X" или "O");
3. Написать функцию для проверки выигрышной комбинации на игровом поле;
4. Добавить возможность завершения игры и объявления победителя или ничьей;
5. Включить в код проверку на невозможность хода

**Ход выполнения**

1. Создаем пустую доску board из двумерного списка
2. Реализуем функцию board\_print() для вывода доски в консоль
3. Пишем функцию win(board, player) для определения наличия выигрышной комбинации, которая будет принимать в себя доску (board) и текущего игрока (player). В функции реализована проверка на выигрышные комбинации по вертикале, горизонтали, диагонали и обратной диагонали.
4. Задаем переменную player, в которой будем хранить текущего игрока
5. Создаем бесконечный цикл while, в котором выводится доска с помощью функции board\_print(),
   1. С помощью print мы сообщаем текущего игрока
   2. Посредством input задаём значение переменным row и column.
   3. Проверяем, допустимы ли координаты.
   4. Проверяем, является ли введенные координаты свободными
   5. Проверяем текущего игрока на выигрышные комбинации при помощи функции win(board, player), если он победил, то сообщаем ему об этом и прерываем цикл.
   6. Меняем текущего игрока на противоположного.

**Листинг программы**

board = [

['-','-','-'],

['-','-','-'],

['-','-','-']

]

def board\_print(board):

for row in board:

for element in row:

print(element, end=' ')

print()

def win(board, player):

for row in board:

if row.count(player) == 3: return True

for i in range(3):

if board[0][i] == player and board[1][i] == player and board[2][i] == player: return True

if all([board[i][i] == player for i in range(3)]): return True

if all([board[i][2 - i] == player for i in range(3)]): return True

player = '0'

while True:

board\_print(board)

print(f"Ход игрока {player}")

row = int(input("Введите строку: ")) - 1

column = int(input("Введите столбец: ")) - 1

if (row > 2) or (row < 0) or (column > 2) or (column < 0):

print('ошибка координат')

continue

if board[row][column] != '-': continue

board[row][column] = player

if win(board,player) == True:

print(f"Победил игрок {player}")

break

if player == '0':

player = 'X'

else: player = '0'

На рисунке 1 представлен вывод программы:

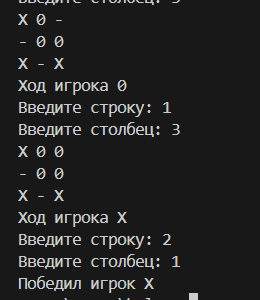
****

Рис. 1 – Вывод программы

**Вывод**

В ходе выполнения лабораторной работы удалось реализовать игру «Крестики-нолики». Это было сделано при помощи следующих шагов:

1. Создали функцию для отображения игрового поля;
2. Реализовали функцию для определения выигрышной комбинации;
3. Ввели механизм для хранения информации о текущем игроке;
4. Добавили отображение победителя, если он имеется;
5. Включили в код проверку на невозможность хода.