ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ

ГОУ ВПО НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ

ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

им. Р.Е. АЛЕКСЕЕВА

ИНСТИТУТ РАДИОЭЛЕКТРОНИКИ И ИНФОРМАЦИОННЫХ

ТЕХНОЛОГИЙ

Кафедра "Вычислительные системы и технологии"

**ПРОГРАММИРОВАНИЕ**

**Курсовая работа  
“Крестики-нолики на большом поле”**

Выполнил студент группы 22-ИВТ-3

Артамонов Никита Александрович

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_«\_\_\_» \_\_\_\_\_\_\_ 20\_\_\_ г. (личная (личная подпись) (дата)

Провел старший преподаватель кафедры «Вычислительные системы и технологии»

Мартынов Дмитрий Сергеевич

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_«\_\_\_» \_\_\_\_\_\_\_ 20\_\_\_ г. (лич (личная подпись) (дата)

Нижний Новгород 2023

**СОДЕРЖАНИЕ**

|  |  |
| --- | --- |
| Введение ……………………………………………………………………  Глава 1. Разработка технического задания ………………………………  Глава 2. Анализ предметной деятельности, уточнение спецификаций и разработка структурной схемы ………………………………………….  Глава 3. Проектирование механизмов игры…………………………….  Глава 4. Проектирование интерфейсов приложения …..………………...  Заключение …………………………………………………………………  Библиография ……………………………………………………………… | 2  6  8  18 |

ВВЕДЕНИЕ

История вычислительных машин и программирования начинается ещё в 19 веке, когда математик Чарльз Бэббидж разработал проект аналитической машины, предназначенной для автоматической обработки математических вычислений. Однако, из-за технических и финансовых проблем, машина не была завершена.

В 20 веке появились первые электромеханические устройства для вычислений, такие как машина Атанасоффа и машина Тьюринга, которые использовались для решения математических задач. В 1941 году Конрад Цузе создал первый программируемый компьютер Z3, который был построен на электронно-механических компонентах.

В 1945 году Джон фон Нейман сформулировал концепцию хранения программ на компьютере, что стало основой для современного программирования. В 1946 году был создан первый электронный компьютер ENIAC, который был гигантским устройством, состоящим из 18 000 ламп и занимающим целый зал.

В 50-х годах появились первые языки программирования, такие как Fortran и Cobol, которые позволяли программистам создавать компьютерные программы на более высоких уровнях абстракции. В 60-х и 70-х годах были разработаны компьютеры с микропроцессорами, что ускорило процесс создания и развития персональных компьютеров. Сегодня мы имеем множество языков программирования и мощные компьютеры, которые позволяют обрабатывать огромные объемы данных и выполнять сложные вычисления. Кроме того, с развитием компьютерной технологии появились новые области, такие как искусственный интеллект, машинное обучение и интернет вещей, которые расширяют возможности использования компьютеров в жизни.

В 80-х годах произошло значительное сокращение размеров компьютеров и повышение их производительности. Это привело к широкому распространению персональных компьютеров в домашней среде, а также к активному использованию компьютеров в бизнесе и научных исследованиях.

В 90-х годах появился Интернет, что изменило не только способы общения людей, но и способы взаимодействия компьютеров. С развитием Интернета появилась возможность удаленной работы, облачных вычислений, электронной коммерции и многого другого.

Сегодня технологии компьютеров и программирования продолжают развиваться быстрыми темпами. Они имеют все большее значение в нашей жизни, используются в различных отраслях, от медицины до промышленности, и открывают новые возможности для решения сложных проблем и создания инноваций. Важно отметить, что в связи с этим компьютерная грамотность становится все более важной для общества в целом, поскольку это позволяет людям более эффективно использовать компьютеры в своих повседневных задачах и профессиональной деятельности.  
История компьютерных игр началась еще в 50-х годах прошлого века, когда на основе электронных вычислительных машин появились первые игровые устройства, такие как "OXO", "Tennis for Two" и "Spacewar!". Однако, настоящий бум игр пришелся на 70-е годы, когда появились первые домашние компьютеры.

В 1972 году появилась первая коммерческая игра "Pong", созданная компанией Atari. Это была простая игра в теннис на двоих, но она стала бестселлером и заложила основу для развития игровой индустрии. Затем следовали другие хиты, такие как "Space Invaders", "Pac-Man", и "Donkey Kong".

В 80-х годах компьютерные игры стали более сложными и многоуровневыми. Появились новые жанры, такие как ролевые игры, стратегии и симуляторы. В 90-х годах произошло поколенное изменение в индустрии, когда появились игры с трехмерной графикой, такие как "Doom", "Myst" и "Quake".

Сегодня игровая индустрия продолжает развиваться с убойной скоростью. Игры выходят на самые разные платформы: компьютеры, консоли, мобильные устройства, а также виртуальная реальность становится все более популярной. Игровые проекты становятся все более сложными и реалистичными. Они предлагают игрокам не только развлечение, но и обучение, социализацию и даже начисление криптовалюты. Некоторые проекты игровой индустрии становятся настоящими масштабными событиями, привлекающими тысячи зрителей и участников со всего мира.  
Игра крестики-нолики является одной из самых старых и популярных игр в компьютерной индустрии. Она была написана еще в 1952 году программистом Клаудием Шенаном.

С тех пор крестики-нолики стали одной из базовых игр, которые поставляются с большинством операционных систем и консолей. Игра имеет несколько вариантов, от простейшего до более сложных и многоуровневых.

Сегодня существуют множество компьютерных игр на основе крестиков-ноликов, которые используют различные правила и графические элементы. В некоторых играх добавлены новые символы, такие как звездочки, сердца или другие, вместо обычных крестиков и ноликов. Также существуют игры на большем поле, где игрокам предстоит выстраивать не только три символа в ряд, но и более, что делает игру более сложной и интересной.

В целом, крестики-нолики и его вариации продолжают оставаться популярными среди любителей компьютерных игр во всем мире.  
Некоторые разработчики компьютерных игр на основе крестиков-ноликов используют искусственный интеллект, чтобы создать более высокий уровень сложности игры. Это позволяет компьютеру адаптироваться к игровому стилю каждого игрока и создавать новые стратегии для победы.

Кроме того, крестики-нолики также часто используются в качестве образовательного материала в школах и университетах, чтобы помочь детям и студентам развивать логическое мышление и умение принимать быстрые решения. Несмотря на то, что крестики-нолики является простой игрой, она по-прежнему остается интересной и увлекательной для многих игроков. Более того, благодаря своей доступности и широкой распространенности, эта игра продолжает привлекать новых поклонников во всем мире.

ГЛАВА 1. РАЗРАБОТКА ТЕХНИЧЕСКОГО ЗАДАНИЯ

1 НАИМЕНОВАНИЕ РАЗРАБОТКИ

«Игра крестики-нолики на большом поле»

2 ОСНОВАНИЕ ДЛЯ РАЗРАБОТКИ

Система разрабатывается в качестве курсовой работы по курсу «Программирование» в соответствии с учебным планом.

3 НАЗНАЧЕНИЕ

Основным назначением программы является проведение досуга с друзьями. Программа может быть использована в качестве базовой модели при создании подобного класса программ.

4 ТРЕБОВАНИЯ К ПРОГРАММНОЙ СИСТЕМЕ

4.1. Функциональные требования:

- Игроки должны иметь возможность ставить свои символы на поле;

- Система должна проверять правильность хода и определять победителя;

- Пользователь должен иметь возможность начать новую игру или продолжить старую;

- Система должна поддерживать игру между двумя игроками, а также игру с компьютером.

Нефункциональные требования:

- Программная система должна быть безопасной, защищенной от несанкционированного доступа;

- Система должна быть быстрой и отвечать на запросы пользователей в режиме реального времени;

- Система должна быть надежной и не выходить из строя при неожиданных входных данных или ошибки в программном коде.

4.2. Требования к надежности

4.2.1 Предусмотреть контроль вводимой информации.

4.2.2 Предусмотреть блокировку некорректных действий пользователя при работе с системой.

4.3 Требования к составу и параметрам технических средств

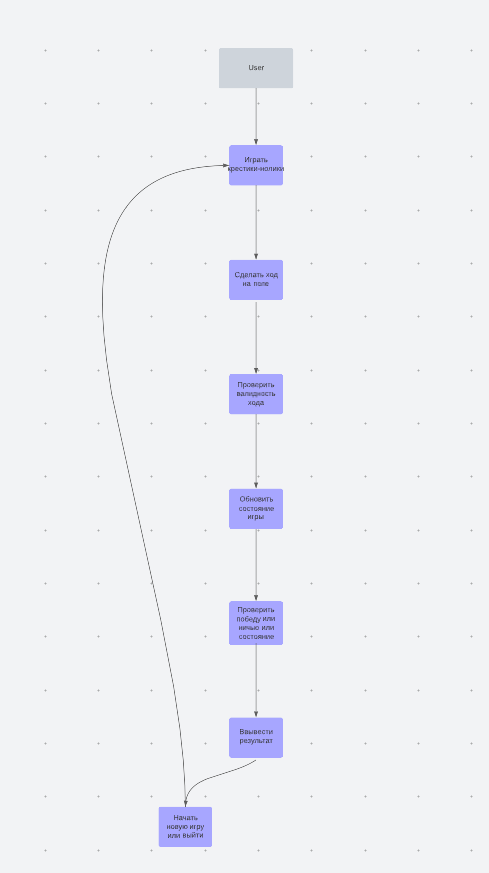
4.3.1 Программная система должна работать на персональных компьютерах типа IBM PC.

4.3.2 Минимальная аппаратная конфигурация:

* ПК на базе компьютера Intel Pentium, оснащенный ОЗУ объемом не менее 128Мб, графическим видеоадаптером, одерживающим видеорежим с разрешением 800х600 и более, с подключенным к нему графическим дисплеем.
* Установленная на ПК операционная система должна иметь возможность запуска Python-интерпретатора (семейство ОC MS Windows XP и выше, Linux/Unix, Mac OS)
* Наличие подключенного к ПК ручного манипулятора «мышь».

ГЛАВА 2. АНАЛИЗ ПРЕДМЕТНОЙ ДЕЯТЕЛЬНОСТИ, УТОЧНЕНИЕ СПЕЦИФИКАЦИЙ И РАЗРАБОТКА СТРУКТУРНОЙ СХЕМЫ

Рассмотрим типичный ход событий в игре крестики-нолики. Для этого используем диаграмму вариантов использования (Use case diagram), где будут представлены основные функциональные возможности системы.



Рассмотрим типичный ход игры:

1. Пользователь запускает приложение;

2. Пользователь выбирает опцию "Играть в крестики-нолики";

3. Система выводит пустое поле и сообщение "Ходит игрок X";

4. Пользователь выбирает ячейку на поле для своего символа (X или O);

5. Система проверяет корректность хода (наличие свободной ячейки и соответствие очередности игрока);

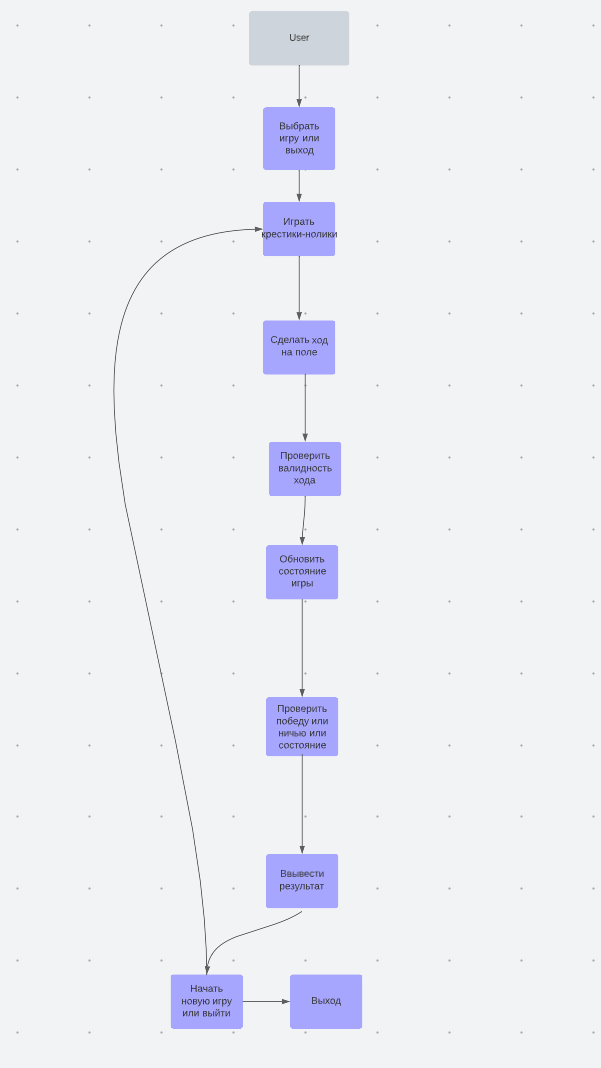
6. Если ход корректен, система обновляет игровое поле и проверяет текущее состояние игры (победа одного из игроков, ничья или продолжение игры);

7. Если игра завершена, система выводит результаты и предлагает начать новую игру или выйти из приложения;

8. Если игра продолжается, система переключает игрока и повторяет процесс с пункта 4.

Таким образом, пользователь может сыграть в крестики-нолики, сделав ход за ходом, до тех пор, пока не будет объявлен победитель, ничья или пользователь не решит выйти из игры.

Рассмотрим альтернативные ходы событий в игре крестики-нолики. Для этого также используем диаграмму вариантов использования (Use case diagram), где будут представлены основные функциональные возможности системы.



Рассмотрим альтернативные ходы игры:

1. Пользователь запускает приложение;

2. Пользователь выбирает опцию "Играть в крестики-нолики";

3. Система выводит пустое поле и сообщение "Ходит игрок X";

4. Пользователь выбирает ячейку на поле для своего символа (X или O);

5. Система проверяет корректность хода (наличие свободной ячейки и соответствие очередности игрока);

6. Если ход корректен, система обновляет игровое поле и проверяет текущее состояние игры (победа одного из игроков, ничья или продолжение игры);

7. Если игра завершена, система выводит результаты и предлагает начать новую игру или выйти из приложения;

8. Если пользователь выбирает начать новую игру, система перезапускает игру с пустым полем и сообщением "Ходит игрок X" (возвращаемся к пункту 4);

9. Если пользователь выбирает выход из приложения, система заканчивает игру.

Таким образом, пользователь может сыграть в крестики-нолики и выбрать между продолжением игры, началом новой игры или выходом из приложения. Цикл игры может повторяться неограниченное количество раз до тех пор, пока не будет принято решение выйти из приложения.

Рассмотрим предметную область приложения.

Предметной областью Крестики-нолики на большом поле является игры на доске, которая представляет собой квадратную матрицу, состоящую из ячеек. Игроки (обычно двое) по очереди ставят на доске крестики или нолики, чтобы получить линию из трех своих символов в любом направлении (вертикально, горизонтально или по диагонали). Однако, в Крестики-нолики на большом поле, доска представляет собой матрицу из меньших досок, и каждый ход игрок должен указать, в какую из этих досок он хочет поставить свой символ. Побеждает тот игрок, которому удается выиграть в достаточном количестве маленьких досок, чтобы заполнить три доски в ряд. Весь процесс обычно происходит на компьютере или на бумаге в виде ручной игры.  
В предметной области Крестики-нолики на большом поле также могут быть определены различные стратегии и тактики игры, которые помогают игрокам достигать победы. Например, игрок может стараться блокировать ходы противника, создавать ловушки и защитные барьеры, или использовать зажимы для захвата нескольких досок сразу. Также в этой предметной области могут использоваться различные алгоритмы и методы искусственного интеллекта для создания компьютерных противников, которые могут играть на разных уровнях сложности и приспосабливаться к стилю игры игрока. Кроме того, существует множество различных вариаций и модификаций этой игры, которые могут изменять правила и усложнять задачу игрокам. В целом, предметная область Крестики-нолики на большом поле представляет собой интересную область исследования для программистов и математиков, которые могут использовать ее для создания новых алгоритмов и вычислительных моделей, а также для разработки новых игр и приложений.  
Помимо того, что Крестики-нолики на большом поле являются игрой, она может быть использована в качестве учебной задачи для изучения программирования, математики и логики, так как в ней содержится множество интересных и сложных задач, требующих использования различных алгоритмов и стратегий.

Предметная область Крестики-нолики на большом поле на C++ представляет собой создание программы, которая позволяет пользователю играть в Крестики-нолики на доске большего размера. Для реализации этой программы необходимо знание основных концепций языка программирования C++, таких как работа с массивами и циклы, использование библиотек для работы с графическим интерфейсом пользователя.

Основными задачами, которые необходимо решить, при разработке программы, являются:

1. Реализация игровой логики, которая позволяет определить текущее состояние игры и проверить, есть ли выигрышное состояние для любого игрока.

2. Визуализация игровой доски и символов игроков, которые были поставлены на доске.

3. Разработка логики пользовательского интерфейса, который позволяет игрокам выбирать ходы и управлять игровым процессом.

4. Тестирование и отладка программы, что позволяет выявить и исправить ошибки в работе программы.

Реализация игры Крестики-нолики на большом поле на C++ может быть использована как для обучения программированию, так и для создания готовых приложений и игр.

Для реализации игры Крестики-нолики на большом поле на C++ можно использовать различные структуры данных, такие как двумерные массивы или векторы, для хранения информации об игровом поле и символах, которые были поставлены на доске.   
Крестики-нолики на большом поле на C++ могут быть расширены и модифицированы, чтобы добавить новые фичи и правила игры. Например, можно добавить несколько игроков или режимы игры с различными уровнями сложности, а также изменить размерность доски или количество символов для победы. Кроме того, игра может быть использована как учебный материал для студентов и программистов, которые стараются развивать свои навыки программирования и создавать собственные игры и приложения.

Давайте уточним спецификации игры Крестики-нолики на большом поле и напишем структурную схему в текстовом формате.

1. Поле имеет размерность N \* N, где N - целое число.

2. Играют два игрока: крестики (X) и нолики (O).

3. Игрок, у которого больше фигур в ряду (по вертикали, горизонтали или диагонали) выигрывает.

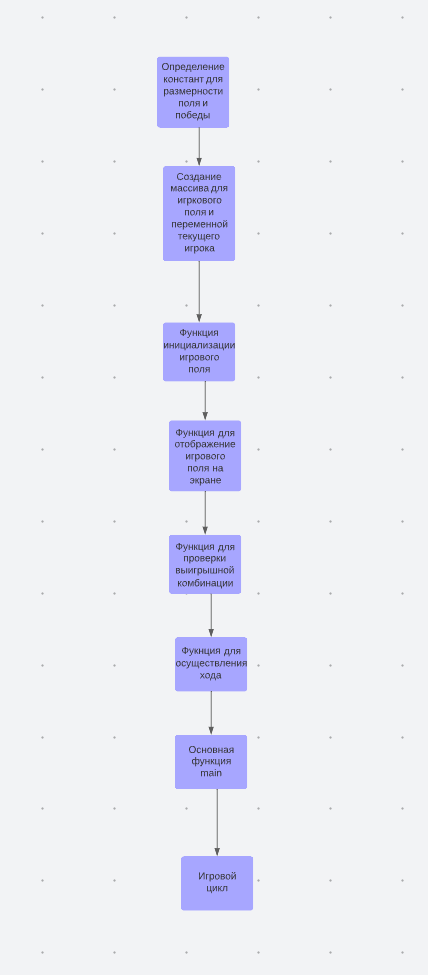
4. Игроки ходят по очереди, начиная с игрока, который играет крестиками.

5. Игрок ставит свою фигуру в любую свободную ячейку поля.

Структурная схема игры Крестики-нолики на большом поле на C++

1. Определение констант для размерности поля и количества фигур в ряду для победы.
2. Создание массива для представления игрового поля и переменной для хранения текущего игрока
3. Функция для инициализации поля.
4. Функция для отображения поля на экране.
5. Функция для проверки выигрышной комбинации.
6. Функция для осуществления хода.

7. Основная функция main, которая будет содержать весь игровой процесс.

Структурная схема в виде блок-схемы:  
  
ГЛАВА 3. ПРОЕКТИРОВАНИЕ МЕХАНИЗМОВ

Реализация крестики-нолики на C++ без использования классов может быть достигнута путем создания функций, которые представляют различные аспекты игры и взаимодействия между пользователем и компьютером. Также необходимо использовать массивы, структуры и указатели для хранения информации об игровом поле и символах игроков. Ниже приводится возможная структура функций для реализации игры.

1. Функция создания игрового поля (InitBoard), которая инициализирует массив, в котором будет храниться информация об игровом поле.

2. Функция отображения игрового поля (PrintBoard), которая отображает текущее состояние игрового поля в консоли.

3. Функция выбора хода игрока (PlayGame), которая запрашивает координаты клетки, в которую игрок хочет поставить свой символ.

4. Функция проверки выигрыша (CheckForWin), которая проверяет, есть ли выигрышное состояние для любого из игроков или ничья.

1. Функция управления игрой (PlayGame), которая определяет, кто ходит следующим, обрабатывает ввод пользователя и уведомляет о выигрыше или ничьей в игре.

Вся эта структура может быть реализована с использованием стандартных библиотек C++, таких как string и iostream, а также простых функций работы с массивами и указателями.

ГЛАВА 4 ПРОЕКТИРОВАНИЕ ИНТЕРФЕЙСА ПОЛЬЗОВАТЕЛЯ

Интерфейс будет состоять из двух частей: отображения игрового поля и запроса хода у игроков. Отображение игрового поля будет осуществляться с помощью вывода на консоль символов, которые представляют крестики, нолики и пустые клетки. Запрос хода будет производиться с помощью ввода данных с консоли.

Для проектирования интерфейса крестики-нолики на большом поле на c++ можно использовать следующий подход:

1. Определить размеры поля и количество клеток.

2. Создать двумерный массив для представления игрового поля.

3. Инициализировать массив пустыми значениями.

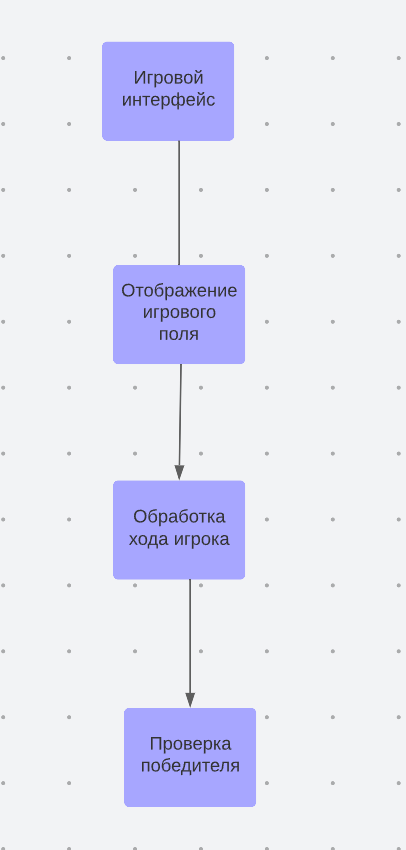
4. Написать функцию, которая будет отображать игровое поле на экране консоли.

5. Написать функцию, которая будет обрабатывать ход игрока и изменять игровое поле.

6. Написать функцию, которая будет проверять наличие победителя.

7. Написать главную функцию, которая будет вызывать остальные функции и управлять игрой.

Для отображения игрового поля мы можем использовать функцию `printBoard()`, которая будет выводить текущее состояние поля на консоль. Для запроса хода у игрока мы можем использовать функцию `playGame`, которая будет запрашивать у игрока координаты клетки, куда нужно поставить его фигуру.

  
  
  
  
ЗАКЛЮЧЕНИЕ

Результатом выполнения курсовой работы стало создание приложения на языке С++. Данное приложение работает в системе с предустановленным ПО. В состав приложения входит игра Крестики-Нолики на большом поле.  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
БИБЛИОГРАФИЯ

Интернет-ресурсы

<https://ru.wikipedia.org/wiki/%D0%9A%D1%80%D0%B5%D1%81%D1%82%D0%B8%D0%BA%D0%B8-%D0%BD%D0%BE%D0%BB%D0%B8%D0%BA%D0%B8>  
<http://absurdopedia.net/wiki/%D0%9A%D1%80%D0%B5%D1%81%D1%82%D0%B8%D0%BA%D0%B8-%D0%BD%D0%BE%D0%BB%D0%B8%D0%BA%D0%B8>  
<https://dzen.ru/a/XKXy-xNNZwCypKdc>  
<https://telegra.ph/Pishem-prostye-krestiki-noliki-na-C-05-17>  
https://www.cyberforum.ru/cpp-beginners/thread1876533.html