Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное учреждение высшего образования

«Новгородский государственный университет имени Ярослава Мудрого»

Кафедра информационных технологий и систем

**Параллельный алгоритм «Гонки»**

Курсовая работа

по дисциплине:

«Операционные системы»

Разработал:

Студент группы 0092

Иванов Н.А.\_\_\_\_\_\_\_\_

«\_\_».\_\_\_\_\_.2022г.

Проверил:

Фуфин М.Н.\_\_\_\_\_\_\_\_

«\_\_».\_\_\_\_\_.2022г.

**Великий Новгород**

**2022**

# **Оглавление**

[**Оглавление** - 1 -](#_Toc104391135)

[**Введение** - 2 -](#_Toc104391136)

[**Глава 1. Разработка проекта.** - 3 -](#_Toc104391137)

[**1.** **Постановка задачи.** - 3 -](#_Toc104391138)

[**2.** **Выбор инструментальных средств.** - 3 -](#_Toc104391139)

[**3.** **Выбор способа взаимодействия потоков.** - 3 -](#_Toc104391140)

[**4.** **Выбор библиотеки для реализации многопоточной модели.** - 4 -](#_Toc104391141)

[**5.** **Алгоритм решения задачи.** - 4 -](#_Toc104391142)

[**6.** **Вывод по первой главе.** - 5 -](#_Toc104391143)

[**Глава 2. Реализация проекта.** - 6 -](#_Toc104391144)

[**1.** **Описание программной реализации.** - 6 -](#_Toc104391145)

[**2.** **Пример работы приложения.** - 6 -](#_Toc104391146)

[**3.** **Результаты работы.** - 6 -](#_Toc104391147)

[**Заключение.** - 8 -](#_Toc104391148)

[**Список используемой литературы.** - 8 -](#_Toc104391149)

[**Макет.** - 9 -](#_Toc104391150)

# **Введение**

Целью данной работы является создание параллельного игрового алгоритма «Гонки», в котором несколько компьютерных игроков соревнуются на поле, где главной задачей является продержаться, как можно дольше.

Особенность проектируемой игры заключается в том, что она должна работать в консоли под операционной системой Windows.

Данная тема может служить показателем производительности при использовании нескольких потоков в разработке игр.

# **Глава 1. Разработка проекта.**

## **Постановка задачи.**

Задачей данной курсовой работы является разработка параллельного игрового алгоритма. Вывод результатов на экран должен осуществляться через консоль. Игра должна работать под управлением операционной системы семейства Windows.

Предполагается, что игра не будет иметь графический интерфейс. В игре будет реализовано предложение выбора: начать игры или закрыть, которое так же выводится в случае поражения игрока.

Правила игры следующие:

1. Предлагается выбор:
   1. Начать игру;
   2. Закончить игру;
2. Пользователь определяет количество игроков от 1 до 10
   1. Меньше 4 – генерируется поле 11 на 11;
   2. Больше 3 – генерируется поле 16 на 16;
3. При столкновении один из игроков выбывает:
   1. В случае столкновения компьютерных игроков – их количество уменьшается;
   2. При столкновении с пользователем – игра заканчивается;
4. Каждые 10 секунд поле игры уменьшается:
   1. Если компьютерный игрок оказывается в удаляемом поле, он выбывает
   2. Если игрок оказывается в удаляющемся поле, игра заканчивается поражением
5. Выбор пользователя:
   1. Начать игру
   2. Закончить игру

## **Выбор инструментальных средств.**

В качестве операционной системы, на которой производилась проверка работоспособности программы, является Windows 10 на архитектуре arm64. Данная система была выбрана из-за суждений распространённости на 2022 год.

Основным языком программирования был выбран C++ из-за личных суждений.

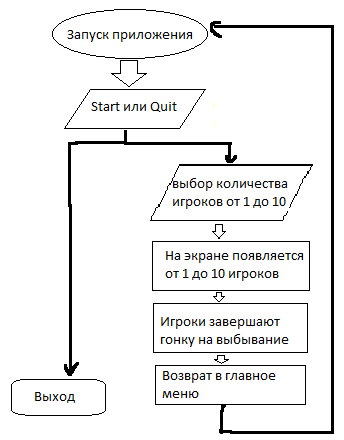
## **Выбор способа взаимодействия потоков.**

При определении способа разделения потоков для выполнения данного алгоритма была выбрана реализация с помощью Мьютексов, так как они просты в использовании и помогают легко обеспечить взаимоисключение выполнения критических секций.

## **Выбор библиотеки для реализации многопоточной модели.**

Для создания потоков и управления их работой существуют различные библиотеки. Для реализации была выбрана стандартная библиотека thread и mutex. Стандартная библиотека C++ предоставляет кроссплатформенные потоки - thread и mutex. Кроссплатформенность нам не понадобится, так как программа работает на системе Windows.

## **Алгоритм решения задачи.**



При запуске приложения появляется консольное окно. На нем присутствует фраза "Press 'Y' to start and press 'N' to close: ". Если ввести “n” или “N”, приложение закрывается. Если ввести “y” или “Y”, выводится следующая фраза "Enter the number of players (from one to ten): ". В зависимости от выбора на экране появляются от одного до десяти компьютерных игроков. Каждым игроком управляет один дополнительный поток. Когда гонка завершается, на экране появляется фраза "Press 'Y' to start and press 'N' to close: " для того, чтобы можно было выбрать начать игру заново или выйти.

## **Вывод по первой главе.**

Таким образом, для реализации параллельного игрового алгоритма «Гонки» основной ОС выбрана Windows, языком программирования C++, без графического интерфейса. Для синхронизации потоков были выбраны мьютексы. В качестве библиотеки, реализующей работу потоков в Windows, был выбран thread.

# **Глава 2. Реализация проекта.**

## **Описание программной реализации.**

В функции main выводится фраза, отвечающая за то, начнётся игра или пользователь хочет выйти. Если выбирается первый вариант – то предлагается выбор количества компьютерных игроков. После этого действия программы переходят в функцию int game(). Переназначаются позиции всех игроков и задаётся размер поля, в зависимости от количества игроков, выбранного пользователем. После объявления ряда переменных запускается цикл на 1000 итераций. Происходит считывание событий (нажатия клавиш), по которому определяется перемещение пользователя по игровому полю. После этого происходит задержка в 200 миллисекунд и условие каждые 10 секунд уменьшать границы поля на единицу и отчищать консоль (функция BOOL exceedsDelay()). Для задержки используется время запуска программы и время системы. Следом происходит перезапись введённых в консоли данных. Указатель смещается в начало и переписывает в консоль поле учитывая перемещение игроков. Запускаются потоки, в которых вызываются случайные числа от 1 до 4, которые характеризуют перемещение компьютерных игроков. Если они сталкиваются или остаются на границе поля во время его уменьшения, то они остаются за игровым полем. Если все противники остаются за игровым полем, то пользователь выигрывает, если он сталкивается с противниками или стоит на границе поля во время его уменьшения – проигрывает. Что в свою очередь возвращает в main() либо 0, либо 1 соответственно.

## **Пример работы приложения.**

Пользователь запускает приложение и вводит один из символов “y” или “Y”. Затем он вводит количество игроков - четыре. Как только пользователь сделал свой выбор, немедленно начинается гонка. Каждые 10 секунд поле игры уменьшается. Когда все компьютерные игроки выбывают из игры, выводится фраза “YOU WON” и пользователя переносит к начальному выбору. Пользователь водит “n” или “N” и выходит из приложения.

## **Результаты работы.**

В ходе выполнения данной курсовой работы был разработан параллельный игровой алгоритм «Гонки», соответствующую поставленным к ней требованиям. Удалось реализовать консольный интерфейс и потоковое чтение нажимаемых клавишь. Игроки только компьютерные. Для обработки хода каждого игрока был использован один отдельный поток. Пользователь может выбрать количество игроков. Вывод результатов гонки на экран производится в основном потоке.

# **Заключение.**

Таким образом, результат курсовой работы соответствует требованиям к реализации. Приложение работает на операционных системах Windows. Для создания приложения был использован язык C++ и компилятор Visual Studio. Многопоточность была реализована с помощью библиотеки thread.

# **Список используемой литературы.**

1. Как написать игру на C++. URL: [Как написать игру на C++ / Хабр (habr.com)](https://habr.com/ru/post/537682/)
2. Цвет текста в консоли. URL: [windows - Цвет текста в консоли - Stack Overflow на русском](https://ru.stackoverflow.com/questions/16431/%D0%A6%D0%B2%D0%B5%D1%82-%D1%82%D0%B5%D0%BA%D1%81%D1%82%D0%B0-%D0%B2-%D0%BA%D0%BE%D0%BD%D1%81%D0%BE%D0%BB%D0%B8)
3. Генерация случайных чисел в диапазоне [a,b]. URL: [алгоритм - Генерация случайных чисел в диапазоне [a,b] - Stack Overflow на русском](https://ru.stackoverflow.com/questions/41317/%D0%93%D0%B5%D0%BD%D0%B5%D1%80%D0%B0%D1%86%D0%B8%D1%8F-%D1%81%D0%BB%D1%83%D1%87%D0%B0%D0%B9%D0%BD%D1%8B%D1%85-%D1%87%D0%B8%D1%81%D0%B5%D0%BB-%D0%B2-%D0%B4%D0%B8%D0%B0%D0%BF%D0%B0%D0%B7%D0%BE%D0%BD%D0%B5-a-b)
4. Мерцание из-за system("cls"). URL: [Мерцание из-за system("cls") - C++ - Киберфорум (cyberforum.ru)](https://www.cyberforum.ru/cpp-beginners/thread1642423.html)
5. Multithreading. URL: [Multithreading / Хабр (habr.com)](https://habr.com/ru/company/otus/blog/549814/)
6. Очистка консоли. URL: [c++ - Очистка консоли - Stack Overflow на русском](https://ru.stackoverflow.com/questions/471914/%D0%9E%D1%87%D0%B8%D1%81%D1%82%D0%BA%D0%B0-%D0%BA%D0%BE%D0%BD%D1%81%D0%BE%D0%BB%D0%B8#comment529516_471916)

# **Макет.**

[Nikolay0092/Kursovay (github.com)](https://github.com/Nikolay0092/Kursovay)