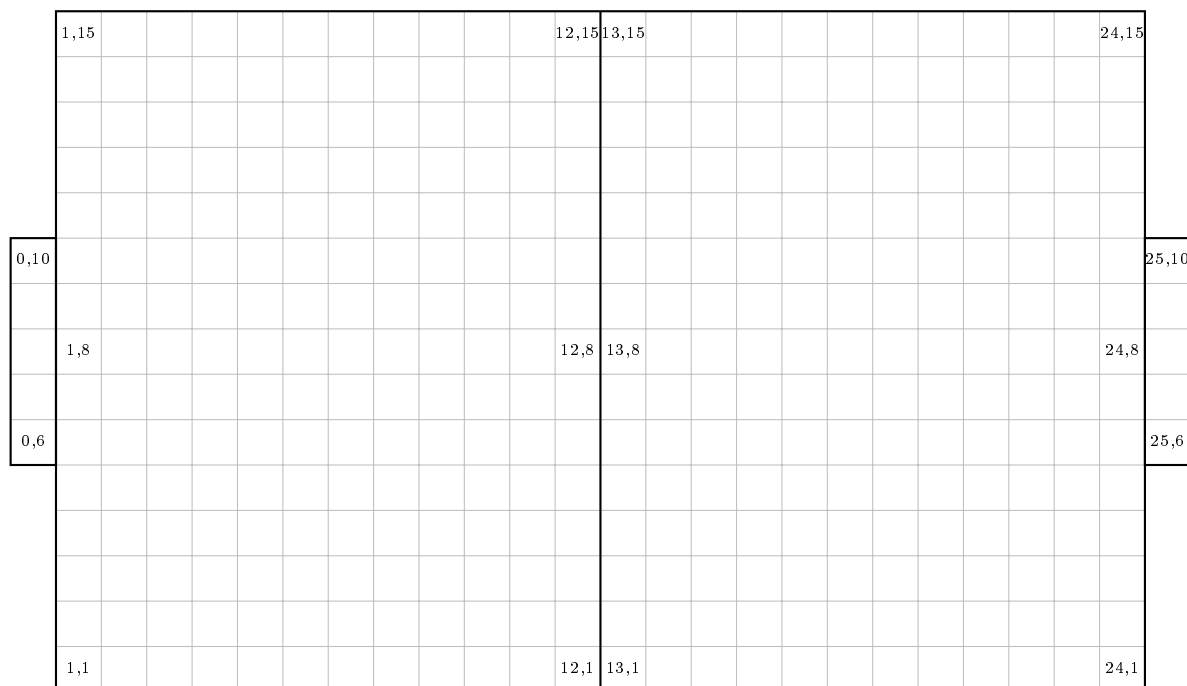


1 Общие положения

1. Вам предстоит написать программу, которая будет играть в футбол против аналогичной программы соперника.
2. Игра проходит на клеточном поле. Игра состоит из двух таймов, каждый из которых длится 45 единиц времени. То есть вся игра длится 90 единиц времени.
3. Каждая из команд состоит из пяти игроков (4 полевых игрока и 1 вратарь).
4. Ваши игроки имеют номера от 1 до 5 включительно (у вратаря номер 5).
5. Игроки соперника имеют номера от 6 до 10 включительно (у вратаря номер 10).
6. Перед каждой единицей времени программа может сделать ход на основе входных данных. Правила ходов и формат взаимодействия с тестирующей системой будут описаны ниже.

2 Схема поля

1. Поле имеет размер 24x15 клеток и делится на две равные половины по 12x15 клеток.
2. За короткими краями каждой стороны по центру расположены ворота, имеющие размер 1x5 клеток.
3. Клетка с координатами (1, 1) расположена в левом нижнем углу поля.
4. На изображении ниже представлена схема поля, на которой отмечены координаты ключевых клеток.



3 Начальная расстановка

1. Каждая из команд владеет одной из половин поля.
2. Между таймами команды меняются половинами поля.
3. Для простоты будем считать, что ваша команда всегда владеет левой половиной поля. Если в текущем тайме ваша команда владеет правой половиной поля, тестирующая система отразит во входных и выходных данных вашу расстановку и ваши ходы.
4. Игроки помещаются в начальную расстановку перед началом каждого тайма, а также после каждого забитого гола (см. п.п. 5).
5. Игрок 1 в начальной расстановке всегда расположен в клетке (12, 8). Если ваша команда разыгрывает мяч, то игрок 1 владеет мячом (см п.п. 7).
6. Игрок 5 (вратарь) в начальной расстановке всегда расположен в клетке (1, 8).
7. Для игроков 2 – 4 вы можете выбрать координаты начальной расстановки во время хода перед первым таймом.
8. Все игроки команды в начальной расстановке должны быть расположены на своей половине поля.
9. Начальная расстановка должна учитывать п.п. 4.2 – 4.3.

4 Состояние игры

1. Каждую единицу времени игра имеет фиксированное состояние, которое задается:
 - координатами каждого из игроков,
 - координатами мяча,
 - одним из следующих двух параметров:
 - вектором скорости мяча, если им никто не владеет,
 - номером игрока, который владеет мячом, иначе.
2. Два игрока одной команды не могут находиться в одной клетке.
3. Игроки не могут находиться в воротах.
4. Мяч не может находиться в воротах, так как сразу после гола применяется п.п. 5.2.

5 Гол

1. Если мяч оказывается в воротах одной из команд, то считается, что противоположная команда забила гол и она получает одно очко.
2. Сразу после гола устанавливается начальная расстановка (см. п.п. 3). Мяч разыгрывает команда, пропустившая гол.
3. Если мяч оказался в воротах в 46-ю или 91-ю единицу времени (после хода на 45-й или на 90-й единице времени), то такой гол тоже засчитывается.

6 Навыки игроков

1. Для всех игроков (включая вратаря) доступны следующие навыки, измеряемые в очках:
 - отбор мяча (t), от 1 до 3 очков (см. п.п. 7.5);
 - скорость (s), от 2 до 3 очков (см. п.п. 8.2);
 - сила удара (p), от 2 до 3 очков (см. п.п. 9.4).
2. Для вратаря доступен навык:
 - отражение удара (c), от 1 до 5 очков (см. п.п. 10)
3. Изначально для каждого игрока установлена минимальное количество очков по каждому из навыков.
4. Перед первым таймом каждой команде выдается 10 очков, которые можно распределить между игроками своей команды, чтобы увеличить их навыки.
5. Количество очков навыка у игрока не может превышать максимальное количество очков этого навыка.
6. Количество очков навыков у игроков противоположной команды становится известно во 2-м тайме.

7 Владение мячом

1. Мячом в единицу времени может владеть максимум один игрок.
2. Если игрок владеет мячом, он находится с ним в одной клетке.
3. Если в одной клетке расположены мяч и единственный игрок, то этот игрок владеет мячом.
4. Если два игрока находятся в одной клетке с мячом, то мячом владеет один из этих игроков. (Более двух игроков в одной клетке быть не может согласно п.п. 4.2).
5. Если выполнены следующие условия:
 - на текущей единице времени игрок А и игрок В находятся в одной клетке с мячом,
 - игрок А имеет t_A очков отбора мяча (см. п.п. 6.1),
 - игрок В имеет t_B очков отбора мяча (см. п.п. 6.1),

то вероятность того, что владение мячом на текущей единице времени перейдет к игроку А, равна

$$P_A = \frac{t_A}{t_A + t_B},$$

а вероятность того, что владение мячом перейдет к игроку В, равна

$$P_B = 1 - P_A.$$

8 Передвижение игроков

1. Игроки могут передвигаться по горизонтали, по вертикали и по диагонали (при движении по диагонали $\Delta x = \pm \Delta y$).
2. Если у игрока s очков скорости (см. п.п. 6.1), то за одну единицу времени игрок может переместиться максимум на
 - s клеток по горизонтали или по вертикали,
 - $s - 1$ клеток по диагонали.
3. Игроки не могут перемещаться за пределы поля, а также нарушать п.п. 4.2 - 4.3.
4. Игрок, владеющий мячом, не может перемещаться вместе с мячом, а также оставаться на месте вместе с мячом.

9 Пасы и удары

1. В рамках нашей игры пас и удар означают одно и то же. Далее будем использовать только слово удар.
2. Игрок, который владеет мячом, может совершить удар.
3. Удары можно совершать по горизонтали, по вертикали и по диагонали (при движении по диагонали $\Delta x = \pm \Delta y$).
4. Если у игрока p очков силы удара (см. п.п. 6.1), то максимальная скорость мяча после удара в клетках на единицу времени равна
 - p , при ударе по горизонтали или по вертикали,
 - $p - 1$, при ударе по диагонали.
5. После удара мяч движется равномерно и прямолинейно.
6. Если мяч при движении после удара достигает краевой клетки поля, не граничащей с воротами, то одна из координат его вектора скорости отражается, в зависимости от того, какой край поля был достигнут. При попадании мяча в угол отражаются обе координаты.
7. При попадании в ворота мяч останавливается.
8. Игрок может одновременно совершить удар и перемещение без мяча (см. п.п. 8.2).
9. Запрещено совершать удар таким образом, что мяч в следующую единицу времени окажется в одной клетке с любым из ваших игроков (за исключением ситуации, когда мяч отражен вратарем, см. п.п. 10). Иными словами, запрещено нарушать п.п. 8.4 и совершать пасы продолжительностью в одну единицу времени.

10 Отражение ударов вратарем

1. Если выполнены следующие условия:

- в единицу времени t вратарь находится в клетке (x, y) ,
- в единицу времени t мяч должен оказаться в воротах согласно своему вектору скорости,
- между единицами времени $t - 1$ (включительно) и t мяч находился в клетке (x, y) ,
- у вратаря s очков отражения удара (см. п.п. 6.2),

то вероятность того, что в единицу времени t мяч окажется в одной клетке с вратарем, равна

$$P = \frac{2c}{10}.$$

2. Если выполнены следующие условия:

- в единицу времени t вратарь находится в клетке (x, y) ,
- в единицу времени t мяч должен оказаться в воротах согласно своему вектору скорости,
- между единицами времени $t - 1$ (включительно) и t мяч находился хотя бы в одной из клеток $(x + \delta_x, y + \delta_y)$, где $0 < |\delta_x|, |\delta_y| \leq 1$, но не находился в клетке (x, y) ,
- у вратаря s очков отражения удара (см. п.п. 6.2),

то вероятность того, что в единицу времени t мяч окажется в одной клетке с вратарем, равна

$$P = \frac{c}{10}.$$

3. Обратите внимание, что п.п. 10.1 – 10.2 не гарантируют переход владения мячом к вратарю после отражения удара. Если в одной клетке с вратарем находится игрок противоположной команды, владение мячом определяется согласно п.п. 7.5.

11 Порядок перехода между единицами времени

1. Все игроки перемещаются на новые позиции.
2. Вычисляется новая позиция мяча с учетом п.п. 9.6 – 9.7.
3. Если новая позиция мяча оказалась в воротах, применяются правила отражения удара вратарем (см. п.п. 10).
4. Мяч перемещается на новую позицию с учетом предыдущего подпункта.
5. Если мяч оказался в воротах, применяются правила гола (см. п.п. 5).
6. Если закончился первый тайм, устанавливается начальная расстановка второго тайма.
7. Если один или два игрока оказались в одной клетке с мячом, определяется владелец мяча (см. п.п. 7).
8. Если закончился второй тайм, игра завершается.

12 Взаимодействие с тестирующей системой

1. Ваша задача — реализовать два метода: `setup_initial_position` и `make_a_move`, описанные в `participant_api.h`.
2. В файле `example_solution.cpp` есть пример решения, в котором также реализованы некоторые другие методы, которые вам могут быть полезны.
3. Все ходы в течение игры запрашиваются у программы в рамках одного запуска, поэтому допускается сохранение необходимых данных между итерациями.
4. Ограничение по памяти для вашей программы — 256 МБ.
5. Ограничение по времени суммарно на все ходы — 5 секунд.
6. Чтобы получить вердикт `Accepted`, программа должна успешно скомпилироваться.

13 Правила турнира

1. Во время проведения номинации решения команд будут соревноваться в небольших турнирах, организованных по швейцарской системе.
2. Турниры во время номинации будут проходить раз в несколько минут, начиная с некоторого момента, когда накопится достаточное количество решений от участников.
3. Перед каждым турниром будут загружаться актуальные решения от участников, имеющие вердикт `Accepted`.
4. Турниры во время номинации будут состоять из шести туров.
5. Если в турнир попало нечетное количество команд, то в каждом туре одна из команд играет против команды жюри.
6. За победу в игре команда получает 3 очка, за ничью — 1 очко, за поражение — 0 очков.
7. Если ход команды нарушает правила игры, программа команды не укладывается в ограничения или завершается с ошибкой, команде засчитывается техническое поражение (со счетом 1:0 в пользу противоположной команды).
8. После завершения номинации будет проведен большой круговой турнир, в ходе которого каждая команда сыграет с каждой два раза.
9. Жюри оставляет за собой право увеличить продолжительность игр в круговом турнире. Опирайтесь в своем коде на константы `GAME_DURATION` и `HALF_DURATION` из `constants.h`.
10. Система начисления очков в круговом турнире совпадает с системой в турнирах по швейцарской системе (см. п.п. 13.6).
11. Победители и призеры номинации будут определяться по итогам кругового турнира.

12. Баллы команды за номинацию будут рассчитываться по формуле:

$$\frac{P}{P_{max}} \cdot 100,$$

где P — количество очков команды в круговом турнире, P_{max} — максимальное среди всех команд количество очков в круговом турнире.

14 Локальная отладка

1. Вам предоставлен пакет участника, который можно загрузить со страницы «Новости».
2. Проект подготовлен для открытия в Visual Studio 2022 (пункт в главном меню Открыть проект или решение, нужно выбрать файл `FootballUserPackage.sln`).
3. Проект можно запускать в четырех режимах: `EmulatorAgainstJury`, `EmulatorAgainstYourself`, `ServerAgainstJury`, `ServerAgainstYourself`.
4. В режимах `Emulator<...>` прогоняются подряд все единицы времени, лог игры записывается в файл `log.json`, который затем можно открыть в визуализаторе (см. п.п. 15.4). В файл `score.txt` записываются количества очков, полученные командами по правилам турнира (см. п.п. 13.6).
5. В режимах `Server<...>` поднимается локальный сервер, к которому может подключиться визуализатор, чтобы запрашивать у вашей программы следующие ходы (см. п.п. 15.5).
6. В режимах `<...>AgainstJury` ваша программа соревнуется с программой жюри, код которой совпадает с изначальной версией файла `example_solution.cpp`.
7. В режимах `<...>AgainstYourself` ваша программа соревнуется сама с собой.
8. Во всех режимах поддержано использование отладчика.
9. Если вы хотите запустить сервер или эмулятор, соревнующий ваше решение с каким-то другим решением, необходимо собрать второе решение в файл формата `.dll` и указать путь до него в Отладка -> Свойства отладки для проекта -> Отладка -> Аргументы команды.

15 Визуализатор

1. К пакету участника прилагается визуализатор.
2. Для запуска визуализатора необходимо запустить файл `wso2025game.exe` (возможно, потребуется проигнорировать предупреждения от антивируса).
3. В визуализаторе доступны два режима: `Upload a file` и `Connect`.
4. В режиме `Upload a file` вы можете открывать json-файлы, полученные в режимах `Emulator<...>` (см. п.п. 14.4).

5. В режиме **Connect** происходит подключение к поднятому локальному серверу, запущенному в режимах **Server**<...> (см. п.п. 14.5).
6. Для отладки своего решения удобнее всего будет поднять сервер, подключиться к нему визуализатором, поставить в коде breakpoints, в визуализаторе — режим «autoplay», и далее отслеживать состояние игры параллельно в отладчике и в визуализаторе.