

# AI Cup 2014

версия документа 2.1

<b>Условия конкурса</b>	<b>2</b>
<b>Игровой мир</b>	<b>2</b>
Общие положения	2
Солдат	3
Танк	3
База	4
Условия победы	4
<b>Технические детали</b>	<b>5</b>
Общие положения	5
Формат запроса	5
Формат ответа	6
Правила обработки запроса	7
Дополнительная информация	7
<b>Песочница</b>	<b>8</b>
Запуск	8
Визуализация	8
Встроенные стратегии	9

## Условия конкурса

Конкурс AI Cup состоится 25 декабря 2014 года, в четверг, с 10:00 до 17:00. Расчет результатов и награждение победителя пройдут сразу по окончании конкурса. Приз - iPad Air 32 Gb - будет вручен участнику, который займет первое место в конкурсе.

Задачей каждого участника будет написать стратегию игры. По завершению конкурса, между стратегиями конкурсантов будет проведен чемпионат по круговой (round-robin) системе. За победу в каждом матче дается 3 очка, за ничью - 1 очко, за поражение - 0 очков. В случае равенства очков, победитель будет определен по результатам личных встреч. Если и это не позволит определить победителя - по договоренности будут проведены дополнительные игры до установления одного победителя.

## Игровой мир

### Общие положения

В игре участвуют два игрока - *красный* и *синий*.

Поле для игры представляет из себя квадрат **24x24** клетки, на котором располагаются *юниты* игроков.



*Начальное расположение юнитов на игровом поле. Система координат игрового поля: ось X слева направо, ось Y сверху вниз. Координата верхней левой клетки - [0, 0].*

Игроки ходят по очереди. В свой ход игрок может либо *переместить* один из своих юнитов на 1 клетку по горизонтали или вертикали (не по диагонали), либо *атаковать* одним из своих юнитов. Первым всегда ходит красный игрок.

При перемещении юнит не может занимать клетку игрового поля, уже занятую другим юнитом (любого игрока) и не может покидать пределы игрового поля.

Каждый юнит имеет конечное количество *хит пойнтов* (hp), которые уменьшаются при атаке на юнит. Когда у юнита заканчиваются хит пойнты, он считается “убитым” и пропадает с игрового поля.

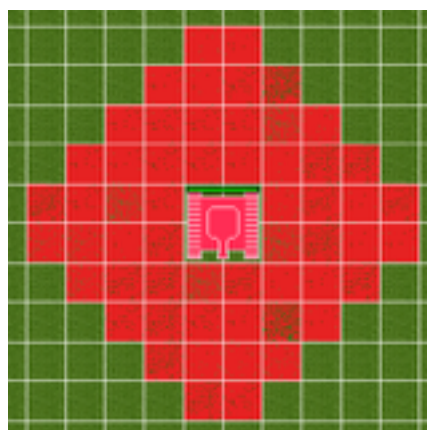
### Солдат



*Солдат синего игрока. Красным отмечены клетки, которые солдат может атаковать.*

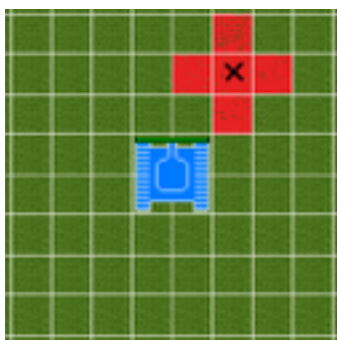
Солдат занимает 1 клетку игрового поля и имеет **15** hp. Солдат может атаковать любую клетку, находящуюся в пределах **2** клеток от себя ([en.wikipedia.org/wiki/Manhattan\\_distance](https://en.wikipedia.org/wiki/Manhattan_distance)) и нанести юниту, занимающему атакуемую клетку урон в **3** hp. После атаки, игрок не может повторно атаковать этим же солдатом в течение **3** своих следующих ходов.

### Танк



*Танк красного игрока. Красным отмечены клетки, которые танк может атаковать.*

Танк занимает на игровом поле квадрат 2x2 клетки и имеет **40** hp. Танк может атаковать любую клетку, находящуюся в пределах **4** клеток от себя и нанести урон в **5** hp атакуемой клетке и ее соседям по горизонтали и вертикали.



*Красным отмечена область поражения при атаке танка на клетку с крестиком.*

Если сразу несколько клеток одного юнита попадают в зону поражения, урон суммируется. После атаки, игрок не может повторно атаковать этим же танком в течение **8** своих следующих ходов.

### База



*База красного игрока.*

База занимает на игровом поле квадрат 3x3 клетки и имеет **100** hp. База не может перемещаться по игровому полю и не может атаковать.

### Условия победы

Игрок считается победителем, если ему удалось уничтожить базу соперника. Если после **2,000** ходов победителя нет, засчитывается ничья.

## Технические детали

### Общие положения

Каждый участник может разрабатывать свою стратегию игры на любом выбранном им языке программирования. Стратегия реализуется в виде http-сервера, который разворачивается участником на любом доступном ему компьютере в локальной сети.

### Формат запроса

При наступлении хода игрока, на сервер с его стратегией будет отправлен http-запрос следующего вида (переносы строк и отступы в теле запроса добавлены для читабельности):

```
POST /path HTTP/1.1
Accept: application/json
Content-Type: application/json
Content-Length: length of request body in bytes
{
  "yourSide": "RED",
  "field": {
    "width": 24,
    "height": 24,
    "units": [
      {"type": "BASE", "side": "RED", "hp": 100, "cooldown": 0, "x": 0, "y": 0},
      {"type": "TANK", "side": "RED", "hp": 40, "cooldown": 5, "x": 4, "y": 4},
      {"type": "SOLDIER", "side": "RED", "hp": 12, "cooldown": 0, "x": 0, "y": 7},
      {"type": "BASE", "side": "BLUE", "hp": 100, "cooldown": 0, "x": 21, "y": 21},
      {"type": "TANK", "side": "BLUE", "hp": 25, "cooldown": 0, "x": 8, "y": 7},
      {"type": "SOLDIER", "side": "BLUE", "hp": 15, "cooldown": 1, "x": 16, "y": 11},
    ]
  }
}
```

Детали:

- метод запроса - **POST**;
- в запросе указаны хэдеры **Content-Type** и **Accept** со значением **application/json**;
- в запросе указан хэдер **Content-Length** со значением, равным длине тела запроса в байтах;
- тело запроса содержит json-объект следующего вида:
  - в поле **yourSide** указан игрок, за которого играет стратегия, возможные значения: **RED**, **BLUE**;
  - в поле **field** описано состояние игрового поля:
    - поля **width** и **height** содержат ширину и высоту игрового поля (в клетках) соответственно;
    - поле **units** содержит массив всех живых юнитов на игровом поле:

- поле **type** содержит тип юнита, возможные значения: **BASE, TANK, SOLDIER**;
- в поле **side** указан игрок, которому принадлежит юнит, возможные значения: **RED, BLUE**;
- поле **hp** содержит текущее количество hp у юнита;
- поле **cooldown** содержит количество ходов, в течение которых юнит не может атаковать;
- поля **x** и **y** содержат координаты левой верхней клетки юнита на игровом поле.

### Формат ответа

В ответ на запрос игрок возвращает свой ход в виде json-объекта в теле успешного (200 OK) http-ответа:

```
HTTP/1.1 200 OK
{
  your move
}
```

У игрока есть три корректных варианта хода:

1. переместить юнит. В этом случае передается json-объект следующего вида:

```
{
  "action": "move",
  "unitX": 4,
  "unitY": 4,
  "direction": "LEFT"
}
```

- 1.1. в поле **action** указывается **"move"**;
- 1.2. в полях **unitX** и **unitY** указывается координата верхней левой клетки юнита, который нужно переместить;
- 1.3. в поле **direction** указывается направление перемещения, возможные варианты: **LEFT, RIGHT, UP, DOWN**;

2. атаковать. В этом случае передается json-объект следующего вида:

```
{  
  "action": "shoot",  
  "unitX":4,  
  "unitY":4,  
  "targetX":5,  
  "targetY":5  
}
```

- 2.1. в поле **action** указывается **"shoot"**;
  - 2.2. в полях **unitX** и **unitY** указывается координата верхней левой клетки юнита, который атакует;
  - 2.3. в полях **targetX** и **targetY** указывается координата клетки, которую нужно атаковать;
3. Пропустить ход. В этом случае передается пустой json-объект: "{}".

### Правила обработки запроса

Максимальное время обработки одного запроса составляет **10 секунд**. Суммарно на игру каждому игроку выделяется **100 секунд** реального времени (в среднем **100 миллисекунд** на ход в случае максимально возможной продолжительности игры). В случае превышения общего лимита все последующие ходы игрок будет пропускать. При замерах потраченного времени будут максимально исключаться затраты на передачу данных по сети.

Если сервер со стратегией не успел обработать запрос, либо вернул некорректный http-ответ (response code, отличный от 200 OK, или неверный формат json) - игрок пропускает этот ход. Если ход игрока некорректен (не обнаружен юнит, юнит принадлежит другому игроку, цель атаки вне досягаемости, юнит не может совершить выбранное действие) - игрок пропускает этот ход.

### Дополнительная информация

Соревнование носит ярко выраженный индивидуальный характер, поэтому совместная работа участников: написание общего кода, коллективное принятие решений, совместное тестирование своих решений - не приветствуется. Мягко говоря.

После сдачи решения, участник не может редактировать его.

В случае проблем с сетью или форс-мажора во время проведения зачетной игры игра будет переиграна после устранения проблемы.

В случае если сервер участника перестал функционировать, участник может перезапустить его перед своей следующей игрой, но не более одного раза за турнир (кроме форс-мажора).

Участник, уличенный в нечестной игре, будет исключен из соревнования.

Решения по всем спорным вопросам принимает организатор, руководствуясь этим документом.

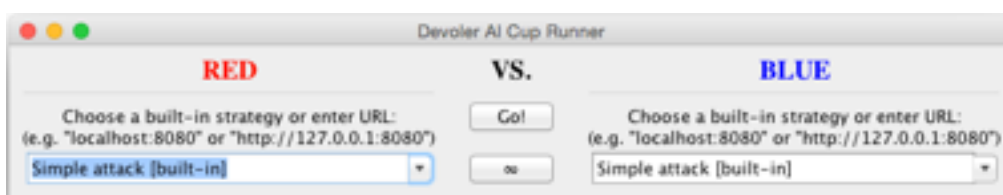
## Песочница

### Запуск

Вместе с этим документом каждому участнику будет выдано приложение “песочница” - AICupRunner.jar. Для запуска приложения:

```
> java -jar AICupRunner.jar
```

Основной экран приложения имеет следующий вид:



Основной экран приложения.

В каждом из двух полей ввода можно выбрать одну из встроенных стратегий, либо указать URL стратегии. После этого необходимо нажать на одну из двух кнопок. Кнопка “Go!” проводит одну игру между выбранными стратегиями (пока идет игра, курсор будет в анимации ожидания), после чего визуально воспроизводит эту игру. Кнопка “бесконечность” проводит игры между выбранными стратегиями до тех пор, пока не будет закрыто диалоговое окно, интерактивно отображающее результаты этих игр.

### Визуализация



Экран визуализации боя.

Экран визуализации боя состоит из трех областей. Основную часть экрана занимает, собственно поле боя, на котором отображается ход игры. Справа от поля боя находится лог



игры, в котором документируются ходы игроков и ошибки. Вверху находится слайдер, управляющий скоростью воспроизведения игры.

### **Встроенные стратегии**

Для удобства участников в “песочницу” встроено несколько стратегий игры:

- *Do nothing* - всегда пропускает ход;
- *Simple attack* - юниты случайным образом движутся к базе оппонента, при возможности атакуя;
- *Simple defence* - юниты остаются на своих стартовых позициях, при возможности атакуя. Если соперник потерял все боевые юниты - переходит в атаку;
- *Balanced* - игрок пытается сбалансированно атаковать и защищаться.

Эти стратегии предназначены для использования в роли несильных и быстрых тестовых оппонентов. Важно понимать, что успех вашего решения против встроенных стратегий никоим образом не гарантирует его успех против решений других участников.