

Midnight Game rules

September 4, 2019

1 Introduction

Tonight you are presented with a unique challenge of creating a bot that will be challenged to play a game we created. The idea is based on a polish contest called PIZZA, which we have come to like and want to present it to you. The game itself is not some math-based one, like you would expect in tasks on Olympic Activities, but rather something you could play online with your friends.

2 The Game

Your task is to control one of several Octocats. In each of following rounds you are matched in groups of 2-4 pairs and put into a 3D universe built of round, shrinking platforms. Your task is to stay alive (that is not fall into the void) for as long as it is possible. To achieve this goal and make it tougher for your opponents you are given two tools:

- Rocket launcher: your character can periodically (with a 2 second cooldown) spawn rockets that travel in a fixed direction for a fixed amount of time. After that they explode pushing the players around them away.
- Black hole launcher: your character can periodically (with a 10 second cooldown) spawn a black hole that will travel in a fixed direction for a fixed amount of time and then stay in place for a few seconds. When in place, it will apply a suCCing force to all the nearby players.

3 Physics

Your character is a point in 3D space with a mass of 1kg, is affected by gravity of $10m/s^2$ and no air resistance. It will collide with platforms underneath it whilst falling on them, but can pass through freely through their bottom. Your character, if on the platform, can declare to **MOVE** in any horizontal direction with a fixed velocity of $5m/s$ or **STOP** in place. That will override its current velocity (yeah, fucc physics).

3.1 Rockets

Rockets are not affected by gravity and other projectiles, and have a fixed speed of $10m/s$. When they explode they instantly change the velocity of all the players by $\frac{1000}{r^2}$, where r is the distance between the rocket and the character in meters, in the direction outwards of the center of explosion. Should they explode **EXACTLY** where some player stands (that is very unlikely with floats, don't try to use it) the explosion is ignored for this character.

3.2 Black holes

Black holes are not affected by gravity and other projectiles, and have a fixed speed of $5m/s$. After stopping, they apply a constant force of $\frac{500}{r^2}$ inwards to its position. Should one exist **EXACTLY** where some player stands, the suCCing force is ignored for this character.

4 Communication

Your program will be communicating with our server using HTTP protocol, so you can create it whatever technology you want to. It is located at `192.168.1.41:80/api` and supports following commands by GET arguments:

4.1 login

Requires following parameters: `nick`, `password`

Returns JSON object of a following structure

```
{
  "Status": 0,
  "Id": "5" //this is your playerId
  "Token": "somalonghex" //use this session token later
}
```

Or a standard error (Section 7)

4.2 getPlayerGames

Requires following parameters: `playerId`

Returns JSON object of a following structure

```
{
  "Status": 0,
  "Games": [1,4,7] //array of gameIds
}
```

Or a standard error (Section 7)

4.3 getPlatforms

Requires following parameters: `gameId`

Returns JSON object of a following structure:

```
{
  "Status": 0,
  "Platforms": //array of platforms
  [
    {
      "X": 5.234,
      "Y": 2.632, //this is the height
      "Z": 9.124,
      "Radius": 2.917,
    }
  ]
}
```

Or a standard error (Section 7)

4.4 getPlayers

Requires following parameters: `gameId`

Returns JSON object of a following structure:

```
{
  "Status": 0,
  "Players": //array of players
  [
    {
      "X": 5.234,
      "Y": 2.632, //this is the height
      "Z": 9.124,
      "VX": 0.345, //velocities
      "VY": -1.432,
      "VZ": -3.123,
      "ID": 5
    }
  ]
}
```

Or a standard error (Section 7)

4.5 getProjectiles

Requires following parameters: `gameId`

Returns JSON object of a following structure:

```
{
  "Status": 0,
  "Projectiles": //array of projectiles
  [
    {
      "X": 5.234,
      "Y": 2.632, //this is the height
      "Z": 9.124,
      "VX": 0.345, //velocities
      "VY": -1.432,
      "VZ": -3.123,
      "Type": 0 //0 for rocket, 1 for black hole
    }
  ]
}
```

Or a standard error (Section 7)

4.6 getMyStats

Requires following parameters: `token`, `gameId`

Returns JSON object of a following structure:

```
{
  "Status": 0,
  "X": 5.234,
  "Y": 2.632, //this is the height
  "Z": 9.124,
  "VX": 0.345, //velocities
  "VY": -1.432,
  "VZ": -3.123,
  "GCool": 0, //time left to use a rocket, 0 if available now
  "BCool": 7.81, //time left to use a black hole, 0 if available now
  "OnGround": 1 //1 if on ground, 0 if not
}
```

Or a standard error (Section 7)

4.7 move

Requires following parameters: `token`, `gameId`, `dx`, `dz`

Moves the player in a given direction, can only be used on ground.

Returns JSON object of a following structure:

```
{
  "Status": 0
}
```

Or a standard error (Section 7)

4.8 stop

Requires following parameters: `token`, `gameId`

Stops the player movement , can only be used on ground.

Returns JSON object of a following structure:

```
{
  "Status": 0
}
```

Or a standard error (Section 7)

4.9 rocket

Requires following parameters: `token`, `gameId`, `dx`, `dy`, `dz`, `time`

Shoots a rocket in a given direction, that will explode after `time` seconds. Has a cooldown of 5 seconds.

Returns JSON object of a following structure:

```
{
  "Status": 0
}
```

Or a standard error (Section 7)

4.10 blackhole

Requires following parameters: `token`, `gameId`, `dx`, `dy`, `dz`, `time`

Shoots a black hole in a given direction, that will stop after `time` seconds and stick around for 2 seconds. Has a cooldown of 15 seconds.

Returns JSON object of a following structure:

```
{
  "Status": 0
}
```

Or a standard error (Section 7)

5 C++ Library

As C++ is (probably) the most popular language among you, we have prepared a library to handle all the network stuff for you. To use it in your program you have to download it from 192.168.1.42/downloads. It contains curl, to communicate with the server, wrapped in some basic functions for each command, a bot template and a Makefile or compile script (depending on the platform). The exact structures and functions it provides are given in its main header file: `game.h`

6 Scoring

Your bot will be scored based on how long it stays in each round. One point is awarded for each second you stay alive in each round. One hundred points are additionally awarded for the player who stays alive in each round the longest. All the points you get in k -th hour of the competition are also multiplied by k .

7 Errors

When you use some command incorrectly, the server responds with a JSON object containing only a nonzero **Status** code. The error codes are defined as follows:

1. Wrong command syntax.
2. Wrong credentials used for login.
3. Game with specified **gameId** does not currently exist.
4. Player with specified **playerId** does not currently exist.
5. You are not in the game with **gameId** given.
6. You tried to move or stop, but you are not grounded.
7. You tried to use grenade/black hole, but it is on cooldown.
8. You sent over 10000 requests in the last 10 seconds. Spam less.
9. The token you specified is not valid for this IP.