INTERN CODING CHALLENGE

Make your mark.

Summer Internship 2015

//RGB WAR

Summer
Internship
2015
//RGB WAR

Reminders

Billing - Please bill your hours to {TREQ 3229927<GO>} for the time that you are in the office during the challenge.

*****Please check in and check out with HR if you decide to leave the building throughout the event, we are in LL2 Germany*****

Help - {MSGE CODING CHALLENGE <GO>} / Coding Challenge IB / Recruiting & Mentors: LL2 Area
June 26th at 9:00pm Physics Office Hour LL2 Canada

Important: Team please let us know if you move locations, this is in compliance with Fire Code Regulations

Food

June 27th at 12:00am	Midnight Fuel	LL2 Pantry
June 27th at 7:00am	Breakfast	LL2 Pantry
June 27th at 12:00pm	Lunch/Judging	28W MPR

We will fill the pantries with snacks, coffee, candy, red bull all night!!!

Rooms

Kick-off, Presentation & Winner Announcement – 28W MPR
Coding Rooms – LL2 Singapore, LL2 UK, LL2 US, LL2 Japan, LL2 Switzerland, LL2 Canada, & LL2 France
Quiet Room – LL2 China
Lounging – LL2 Germany

Who's Here

R&D Recruitment Team:

6:00pm – 1:00am: Amy Anderson, Katie Golub, Kristen Galgano, Willow Marcon, Zoey Deng, Nicole Stelluti
1:00am – 8:00am: Rakhee Kher, Bina Carroll, Jacquie Lennon
8:00am – 4:00pm: Zoey Deng, Lucy Wallace, Wendy Kim, Christina Nguyen

The Crew: Chris Martin, Josh Greenman, James McIlhargey, Aleks Tamarkin, Dan Wilkey

R&D Mentors:

Name	Team	Coverage
Ajmal Sharif	R&D Binary Analytics	Friday 6PM - 1AM
Bharath Reddy Palle	R&D FIT Cash Trading Svcs	Friday 6PM - 1AM
David Baird	R&D Equity Estimate Pub	Friday 6PM - 1AM
Jorge Ornelas	R&D ECD Client tools/Backend	Friday 6PM - 1AM
Nelson Yi	R&D TS AIM Cash/Collateral Mgt	Friday 6PM - 1AM
Peter Soong	R&D Tkr Plant US Equity & Opt	Friday 6PM - 1AM
Peter Szocs	R&D Equity Derivatives Core	Friday 6PM - 1AM
Saurin Kadakia	R&D FX Price Discovery	Friday 6PM - 1AM
Shivashis Saha	R&D TSCI Feeds Framework	Friday 6PM - 1AM
Jitesh Dani	R&D HR	Friday 6PM - 1AM
David Baird	R&D Equity Estimate Pub	Overnight 1AM - 8AM
Jean Manuel Nater	R&D Corps/Capital Markets	Overnight 1AM - 8AM
Peter Martin	R&D Systems Core-New York	Overnight 1AM - 8AM
Sasha Borovinsky	R&D TS TOMS CDS	Overnight 1AM - 8AM
Daniel Henriques	R&D TSCI/TS Systems/Backend	Saturday 8AM - 3PM
David Baird	R&D Equity Estimate Pub	Saturday 8AM - 3PM
Doosan Baik	R&D Feeds - News/Contributions	Saturday 8AM - 3PM
Kevin Johnson	R&D Systems Technology (NY)	Saturday 8AM - 3PM
Rob Ji	R&D TS TOMS Derivatives/MO	Saturday 8AM - 3PM
Yuecheng Shao	R&D Equity Derivatives Core	Saturday 8AM - 3PM

Base Invaders – Game Play

Backstory:
In a distant time in a place far far away, a competition rages amongst the space pilots.
The name of this competition: BaseInvaders
The pilots' play for points, eternal glory and best of all real world prizes.
You own one of the ships, but you are not one of the pilots. You have something better up your sleeve.
You have outfitted your ship with a computer which will execute any program you upload.
It is now your job to write this program that will steer your ship to victory

The Game:
The rules of the game are simple:
There are preplaced markers (codenamed MINE) in space.
If your ship gets in range of one of the MINEs, The MINE is yours until another ship gets in range.
If more than one ship is in range, nobody owns the MINE until only one ship is in range again.
Every predetermined period, a point is added for each mine a ship owns.

After a predetermined time, the game ends and the ship with the most points wins.
In order to achieve victory, your ship has been outfitted with a set of gadgets.
Your ship has a thruster. This thruster can be pointed in any direction, and can be fed any amount of fuel up to a maximum. The laws of physics do apply so any object in motion will remain in motion. There is also a slight amount of kinetic friction. A pilot's manual on steering has been attached to this document.
Your ship is also equipped with brakes.
Your ship has two sensors:
One sensor will scan the immediate area around the ship and return the coordinates of all the objects in range.
The other sensor will return all the objects in range of any position in the world. This sensor needs to be charged to use and therefore has a lengthy cool down.
Your ship has also been equipped with a bomb factory and the ability to place bombs. Due to the weight of bombs, your ship can only be manufacturing/carrying one bomb at a time.
You can place a bomb anywhere in a range. The bomb will explode after a certain time, and push away any ships in the area including yours. The closer the ship is to the center of the explosion, the more it will be pushed away.

With these tools, you have everything you need to achieve victory.
Good Luck.

Game Commands:
Below are the possible commands that you can send to the server. All commands should be followed by a newline. Note that the first line sent to the server constitutes your login. There is no command name, simply send 'USER_NAME PASSWORD', followed by a newline. Only one simultaneous connection is allowed per user name. If you'd like to test multiple versions of your bot in parallel, be sure to use separate logins.

1. STATUS

   Returns – X Y DX DY [MINES] NUM_MINES (OWNER X Y)* [PLAYERS] (X Y DX DY)* [BOMBS] (X Y)*

   The STATUS command provides information concerning all activity within your current viewing radius. That includes your own coordinates and velocity (x, y, dx, dy), all visible gold mines, all visible players (including their instantaneous velocities), and all live bombs. You do NOT know how much time is remaining on a given bomb's fuse.

2. ACCELERATE RADIANS(0 - 2pi) ACCEL(0 - 1)

   Returns – ACCELERATE_OUT

The ACCELERATE command is your primary means of movement. Note that this command does not allow you to set your actual acceleration to a value between 0 and 1, it merely allows you to set your thruster somewhere between minimum (0) and maximum (1) output. For example, if you send the command ‘ACCELERATE 3.14 1’, then your thruster will be set to full at an angle of 180° (relative to the map, not your ship; your ship does not really have its own orientation, and your thrusters can simply be pointed instantaneously in any direction), and you will accelerate to peak velocity as quickly as possible.

3. BRAKE

   Returns – BRAKE_OUT

   This command will bring your ship to a halt as quickly as possible. Note that this command is not strictly necessary. By accelerating at an angle offset pi radians from your current trajectory, you can accomplish the same thing.

4. BOMB X Y or BOMB X Y T

   Returns – BOMB_OUT

   Place a bomb at the specified position, which must be within your viewing radius. You may only place one bomb per second. T is an optional parameter specifying the timer for the bomb in frames.

5. SCAN X Y

   Returns – Same return as STATUS, but centered on arbitrary coordinates instead of your ship’s.
   Can only use once every 5 seconds.

   This is the same as the STATUS command, except you can read status for any (x,y) coordinate in the entire map. Though throttled, you can use this command to scope out areas of the map to use in path planning.

6. SCOREBOARD

   Returns - (PLAYER SCORE MINES_OWNED)*

   As the name implies, this command retrieves the scores of all players currently in the field. Note that it does not tell you the rate at which scores are changing, though it should be possible for you to determine this yourself.

7. CONFIGURATIONS

   Returns - CONFIGURATIONS_OUT
   MAPWIDTH * MAPHEIGHT * CAPTURERADIUS * VISIONRADIUS *
   FRICTION * BRAKEFRICTION * BOMBPLACERADIUS *
   BOMBEFFECTRADIUS * BOMBDELAY * BOMBPOWER * SCANRADIUS *
   SCANDELAY *

   Note that * is a numerical value for each field and that the fields and values are separated only by a single space each. This command provides all of the pertinent game configuration parameters to your bot, as these may not be the same as are set during bot development.

Base Invaders – Local Testing

Obtaining the server jar:
You need three files to run the server, which live in ‘P:\atamarki\BaseInvaders’ on CORP or ~atamark1/BaseInvaders on DEV. You will need the server jar file (BaseInvaders.jar), the settings file (settings.cfg), and the relevant image (spaceships.png). This jar file must be run with Java version 1.8.

Java on Dev:
Java 1.8 is located at /opt/bb/bin/java on all machines.

Java on Corp:
Bring up the software catalog (located under the start menu, just begin typing ‘software catalog’). Navigate to ‘Java Runtime 8 Update 45 – Optional’. Select it and click install. If that version of Java is already installed, it will tell you so, otherwise that will kick off the install.

Java on your laptop:

https://java.com/en/download/

Running the Server:
Create a directory containing both the server jar and configuration file. Run the following command from that directory to bring up the server:

```
java -jar BaseInvaders.jar
```

At this point, the server is running. There are 10 user name / password pairs that are valid to connect to your local server. They are the letters a through j. For example, passing a username of ‘a’ and a password of ‘a’ at startup will allow a connection. Same with ‘g’ and ‘g’, etc. By default, the server will bind to localhost and listen on port 17429. Any of these parameters can be changed by editing the settings.cfg file.

Your server is now ready for testing. The UI should launch so that you can verify it is up. To prove you can connect, you can run, for example:

```
telnet localhost 17429

a a

STATUS
```

That should output a status message for the ship with user name ‘a’. For example, something like this:

```
STATUS_OUT 512.0 512.0 0.0 0.0 MINES 3 -- 542.0150474167419 404.67105520638904 -- 473.2410246299456
423.9064931687125 -- 611.0567644236098 594.879095429467 PLAYERS 9 512.0 512.0 0.0 0.0 512.0 512.0 0.0 0.0 512.0
512.0 0.0 0.0 512.0 512.0 0.0 0.0 512.0 512.0 0.0 0.0 512.0 512.0 0.0 0.0 512.0 512.0 0.0 0.0 512.0 512.0 0.0 0.0 512.0
512.0 0.0 0.0 BOMBS 0
```

Note on Dev Ports:

The server is configured to run on port 17429 by default, as discussed above. If you are bringing up your local instance on a dev machine, be aware that another team may do the same on the same dev machine. In that cause, your sever will fail to bind to the port that is already in use and exit immediately. We encourage you to change your port number in the config file to be unique or just switch machines if you find yourself conflicting with other teams.

That’s it! Happy testing!

Base Invaders – Basic Physics for Pilots

Kinematics

The game you have before you relies upon very basic kinematics dictating how your ship moves through space. In a single direction, motion can be described with two equations:

$$x(t) = x_0 + v(t) \cdot t + \frac{1}{2}a(t) \cdot t^2$$
$$v(t) = v_0 + a(t) \cdot t$$

where x is your position, v your velocity and a is your acceleration. For a computerized, discreet system like the one in this game, we can rewrite these as such:

$$x_t = x_{t-1} + v_{t-1} \cdot \Delta t + \frac{1}{2}a_{t-1} \cdot \Delta t^2$$
$$v_t = v_{t-1} + a_{t-1} \cdot \Delta t.$$

These equations relate your position and velocity at the start of the round, (x_{t-1}, v_{t-1}), to what they will be at the end of the round. A common optimization is to offset the velocity a half step from the position. So first we compute $v_{t+1/2}$:

$$v_{t+1/2} = v_t + a_t \cdot \frac{\Delta t}{2}$$

Then we compute x_{t+1}:

$$\begin{aligned} x_{t+1} &= x_t + v_t \cdot \Delta t + \frac{1}{2}a_t \cdot \Delta t^2 \\ &= x_t + \left(v_{t+\frac{1}{2}} - a_t \cdot \frac{\Delta t}{2}\right) \cdot \Delta t + \frac{1}{2}a_t \cdot \Delta t^2 \\ &= x_t + \ v_{t+1/2} \cdot \Delta t \end{aligned}$$

Then we add another half time step to the velocity:

$$v_{t+1} = v_{t+1/2} + a_t \cdot \frac{\Delta t}{2} = v_t + a_t \cdot \Delta t$$

If we further simplify the equations with $\Delta t = 1$, this leaves:

$$x_t = x_{t-1} + v_{t-\frac{1}{2}}$$

$$v_t = v_{t-1} + a_{t-1}$$

The game is in two dimensions of course, so these need to expand to:

$$x_t = x_{t-1} + vx_{t-\frac{1}{2}}$$

$$y_t = y_{t-1} + vy_{t-\frac{1}{2}}$$

$$vx_t = \mu(vx_{t-1} + ax_{t-1})$$
$$vy_t = \mu(vy_{t-1} + ay_{t-1})$$

where $ax = a \cdot \cos(\theta)$ and $ay = a \cdot \sin(\theta)$, and μ is the friction of space, typically .99.

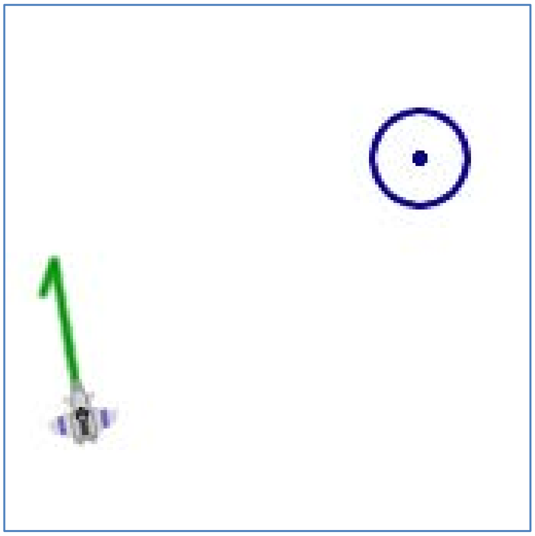
One final note about acceleration: the BRAKE command sets the velocity equations to:

$$vx_t = \vartheta \cdot vx_{t-1}$$
$$vy_t = \vartheta \cdot vy_{t-1}$$

where ϑ is the brake friction, default set to .987.

What’s your vector, Johnny?

As a player, you get to know your position, x, y, your velocity vx, vy, and you get to set your acceleration, a, θ. An essential skill for any pilot is to know how to adjust their heading. Suppose you are in this situation:



You want to adjust your velocity (in green) to point at the nearby mine. Your position is the point (1,1), your velocity is vector $\langle -.520, 2.954 \rangle$, and the mine’s position is point $(10, 9)$.

First thing you want to do is determine the direction you wish to face. Directions are vectors and a vector can be derived from the subtraction of two points:

$$\vec{D} = p_{mine} - p_{player} = \langle 10-1, 9-1 \rangle = \langle 9,8 \rangle$$

Furthermore, we can normalize this by taking the magnitude and dividing it, so it is of length 1:

$$\vec{d} = \vec{D}/_{|D|} = \frac{\langle 9,8 \rangle}{\sqrt{9^2+8^2}} \approx \langle .747, .664 \rangle \approx \langle \cos(41.63°), \sin(41.63°) \rangle$$

Now we want our velocity to point in this direction. First, let’s normalize it to get our angle (where in I cheat because I chose the angle first):

$$\vec{v} = \vec{V}/_{|V|} = \frac{\langle -.520, 2.954 \rangle}{3} = \langle \cos(100°), \sin(100°) \rangle$$

Now we want our new vector to be the same as the direction to the mine. This can be stated as:

$$\vec{V} + \vec{A} = n \cdot \vec{d}, \qquad \vec{A} = \langle A_x, A_y \rangle$$

where n is the new magnitude of our velocity, and can be anything. Letting $n = 3$:

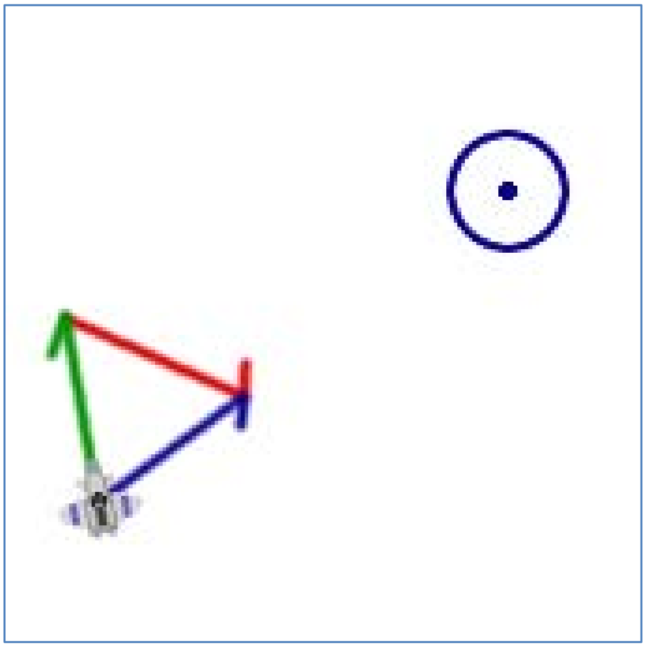
$$3\langle \cos(100°), \sin(100°) \rangle + \langle A_x, A_y \rangle = 3\langle \cos(41.63°), \sin(41.63°) \rangle$$
$$\to A_x = 3\cos(100°) - 3\cos(41.63°) = 2.76194$$
$$\to A_y = 3\sin(100°) - 3\sin(41.63°) = -.962423$$

So that our desired acceleration is:

$$\vec{A} = \langle A_x, A_y \rangle = \langle 2.76194, -.962423 \rangle$$
$$\approx 2.923 * \langle \cos(-19.211°), \sin(-19.211°) \rangle$$

This is seen in the new picture to the right; our current velocity in green, the velocity we want to go in blue, and the acceleration in red.

However, as you may have noticed, the acceleration has a magnitude of 2.923 (by design). So in this game, you cannot do this one time step. We will leave it to you to figure out where to go from here, but all the basic mathematics you will need is above.

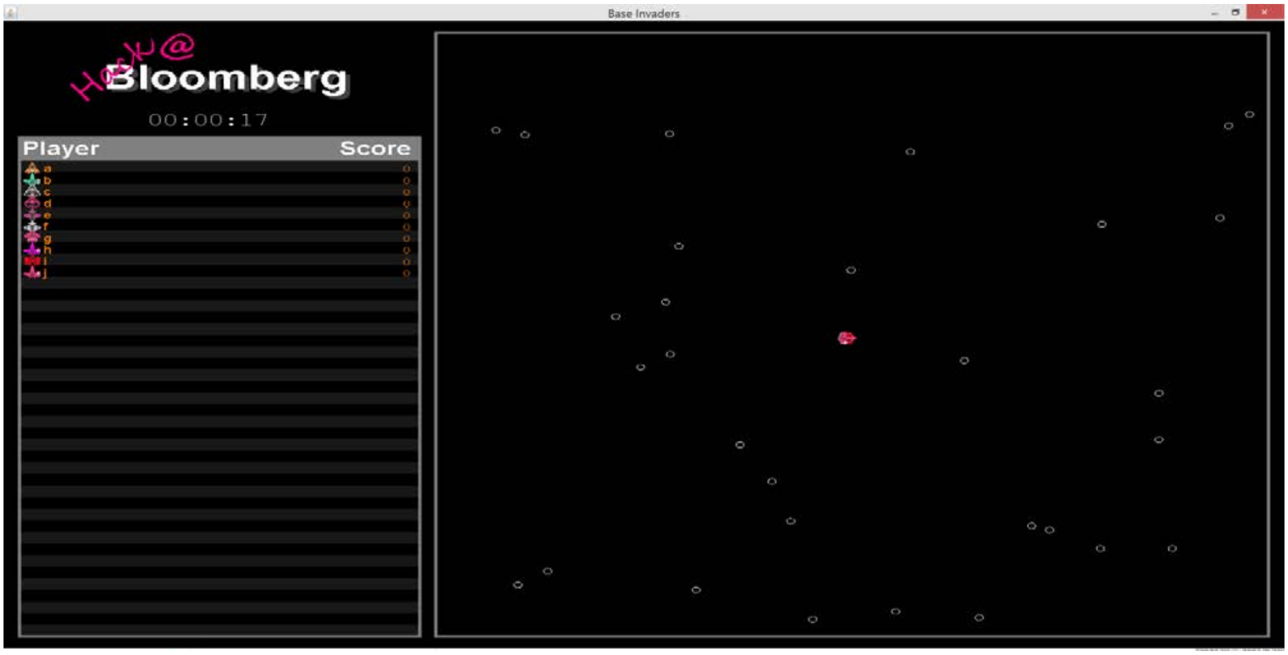


Base Invaders – UI Contest

UIs are about giving out information in an easily digestible manner that is both aesthetically pleasing and serves a purpose, like a massive hot pink banner that grabs your attention!

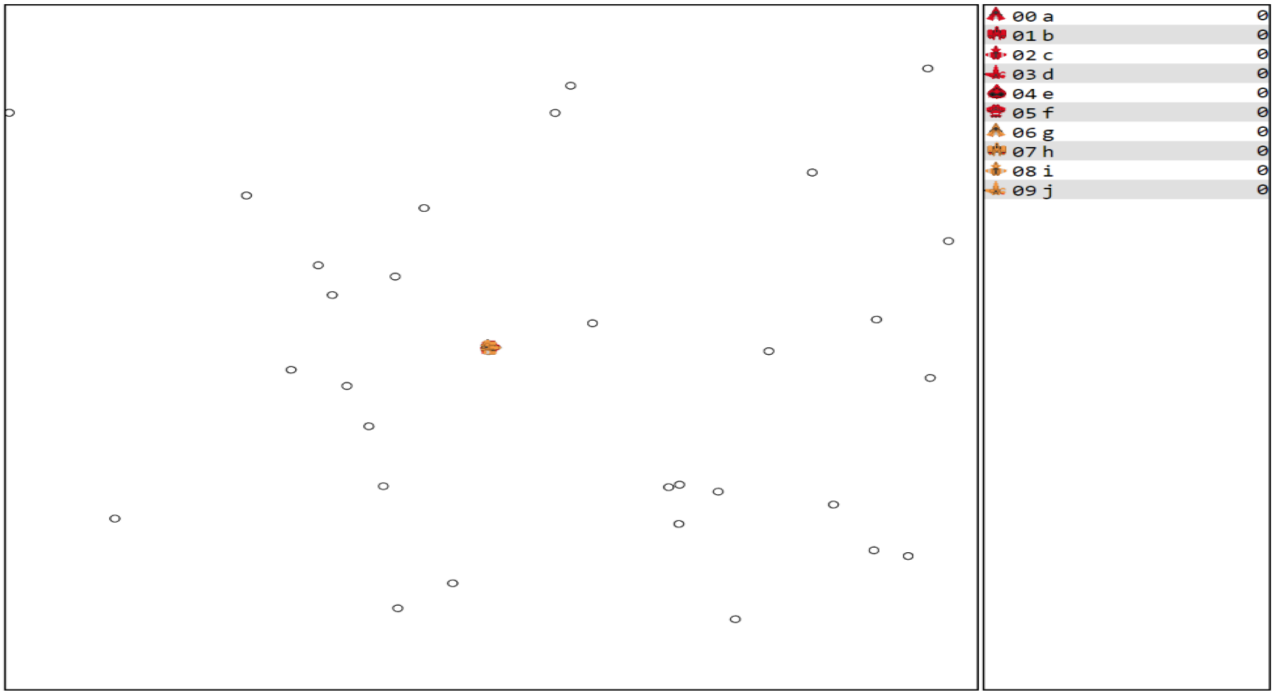
With that in mind, part of this competition will be about feeding up SOMETHING to a user in a potentially interactive manner. That something could be an overhead view of the current state of the game.

You could improve upon the UI for the standalone mode:



Player	Score
a	0
b	0
c	0
d	0
e	0
f	0
g	0
h	0
i	0
j	0

Maybe you're a fan of the web version:



00 a	0
01 b	0
02 c	0
03 d	0
04 e	0
05 f	0
06 g	0
07 h	0
08 i	0
09 j	0

Remember, the prizes for this UI portion are identical to those for the algorithmic portion. Don't be afraid to get creative and think of something completely new…

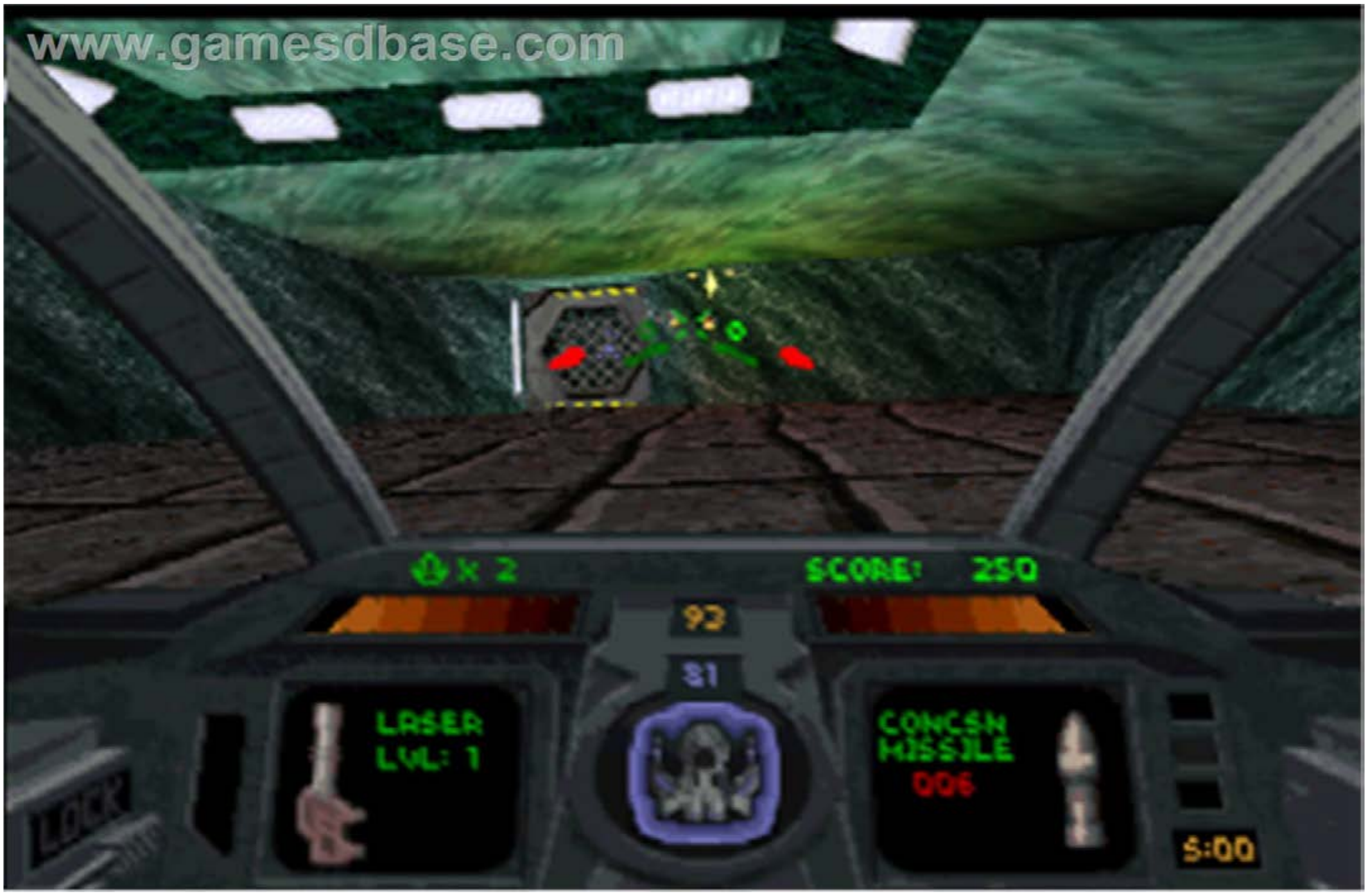
BUT WAIT!

THERE IS SO

MUCH

MORE!!!

Instead of a birds-eye view of the game, you might want to make your ship interactive and playable like an old-school classic Descent game.



Maybe you’re more a fan of exploring the map so that the rest of your imaginary ships will know where all the mines are, just like an overlord in StarCraft. You could click all around a game board to reveal what is actually there while doing almost nothing!

Hey team, Can I float somewhere and look at something for anyone?

What it is and what it becomes is up to you. This is a choose-your-own-adventure book where you get to be the narrator and the protagonist.