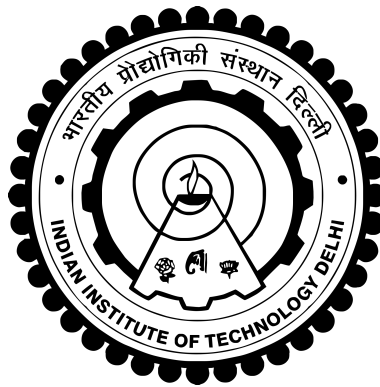


COP 290 Assignment 3

Space Invaders



Faran Ahmad
2013CS10220

Kabir Chhabra
2013CS50287

Kartikeya Gupta
2013CS10231

Prateek Kumar Verma
2013CS10246

March 2015

Contents

1	Objectives	3
2	Overall Design	3
3	Sub Components	3
3.1	Back End	3
3.1.1	Alien	3
3.1.2	Ship	3
3.1.3	Color	3
3.1.4	Bullet	4
3.1.5	Board	4
3.2	Artificial Intelligence	5
3.3	Graphics	5
3.4	Network Part	5
4	Interaction amongst Sub Components	5
4.1	Back-end and UI	5
4.2	Back-end and Network	5
5	Testing Of Components	5
5.1	General Unit Tests	5
5.2	Graphics	5
5.3	Artificial Intelligence	5
5.4	Network Component	5
5.5	Overall Testing	5
6	Extra Features	5
6.1	Competitive Multi-player Mode	5
6.2	3D Game-play	5

1 Objectives

About the assignment

2 Overall Design

1. About the components and layers

3 Sub Components

3.1 Back End

3.1.1 Alien

Listing 1: Class Parameters for Alien

```
1 class Alien
2 {
3     private:
4         float XPos;
5         float YPos;
6         float Angle;
7         Color ColorOfAlien;
8         int Level;
9         int PresentLives;
10        int NumberBullets;
11        int NumberMissiles;
12        int AlienType;
13    };
```

3.1.2 Ship

Listing 2: Class Parameters for Ship

```
1 class Ship
2 {
3     private:
4         float XPos;
5         float YPos;
6         float Angle;
7         std::string Name;
8         Color ColorOfShip;
9         int Lives;
10        int Score;
11        int Multiplier;
```

```

12     int Kills;
13     int Id;
14     int NumberBullets;
15     int NumberMissiles;
16     int AILevel;
17 };

```

3.1.3 Color

Listing 3: Class Parameters for Color

```

1 class Color
2 {
3 private:
4     float R;
5     float G;
6     float B;
7 };

```

3.1.4 Bullet

Listing 4: Class Parameters for Bullet

```

1 class Bullet
2 {
3 private:
4     float XPos;
5     float YPos;
6     float VelX;
7     float VelY;
8     Color ColorOfBullet;
9     int ShipId;
10    bool TypeAI;
11    bool TypePlayer;
12 };

```

3.1.5 Board

Listing 5: Class Parameters for Board

```

1 class Board
2 {
3 private:
4     std::vector<Ship> VectorShips;

```

```

5     std::vector<Bullet> VectorBullets;
6     std::vector<Alien> VectorAliens;
7     double DimensionPosX;
8     double DimensionPosY;
9     double DimensionNegX;
10    double DimensionNegY;
11 };

```

3.2 Artificial Intelligence

3.3 Graphics

3.4 Network Part

4 Interaction amongst Sub Components

4.1 Back-end and UI

4.2 Back-end and Network

5 Testing Of Components

5.1 General Unit Tests

Listing 6: Class Parameters for Test

```

1  class Test
2  {
3  private:
4      bool verbose;           //If test is to be conducted
5      std::string description; //String description of the test
6      bool isPass;           //Boolean if the test has passed
7      void PrintPassFail(bool); //Prints the status of the test
8  };

```

We will use the aforementioned class “Test” to perform unit tests on the different files created. This will ensure that all the functions work correctly against some tests.

5.2 Graphics

5.3 Artificial Intelligence

5.4 Network Component

5.5 Overall Testing

6 Extra Features

6.1 Competitive Multi-player Mode

6.2 3D Game-play