

情報コミュニケーション創成 レポート

クラス B1 (グループ D)

ZHENG Yuhui - 2600120572-1

提出日:2015 年 7 月 20 日

目 次

1 前書き	2
2 プログラムの概要	2
3 外部仕様	2
3.1 起動方法	2
3.2 操作方法	2
3.3 画面遷移	3
4 内部仕様	3
4.1 クラス図	3
4.2 各クラスの説明	4
5 実行例	5
5.1 ディスクトップ	5
5.2 プログラム起動した直後	5
5.3 ゲーム画面	6
5.4 ゲームオーバー	7
6 工夫した点	8
6.1 プログラム上	8
6.2 デザイン上	8
6.3 チームワーク	9

1 前書き

今回のワークショップの操作環境はいろいろ限定があるので、予想のように面白いプログラムを実装することはかなり難しいと思う。コマンドプロンプトを UI として実装されたソフトはハイライト少ないと想いながら、ブラウザを使い、グラフィックスデザインインターフェイスを実装したゲームの方は迫力的であるじゃないでしょうかと考える。今回はこのように jquery.js を使ってブラウザで実行するゲームを開発した。

2 プログラムの概要

AirForce は、簡単な 8bit スタイルの垂直バージョン・シューティングゲームである。キーボードで戦機をコントロールし、弾を打ち、敵を倒すことができる。

3 外部仕様

3.1 起動方法

コンピューターを立ち上げると、ディスクトップにゲームのアイコンをクリックすると、ブラウザが立ち上がる。プログラムが起動したら、ゲームが自動的に始まる。

3.2 操作方法

キーボードでプレーヤーの戦機を制御することができる。

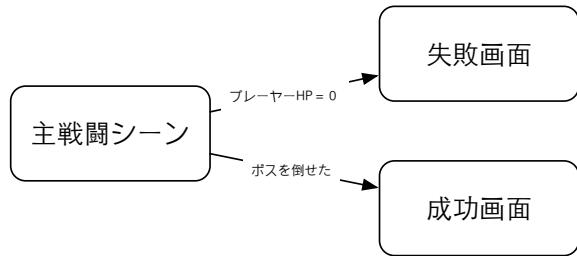
- W:
- S:
- A:
- D:
- J: 発射

戦機は自分のライフ値を持ち、HP は 0になると戦機は消える。プレイヤー戦機のライフ初期値は固定し、敵の HP はレベルによって違う。

敵戦機を倒せる時、成就の評価基準とする点数を得る。最終の目標は、ボスを倒すこと及び更に高い点数を挑戦することである

ある敵戦機を倒せば緑の円の供給ユニットが出る場合がある。このユニットとすることにより、武器をアップグレードしたり、HPを回復したりすることができる。

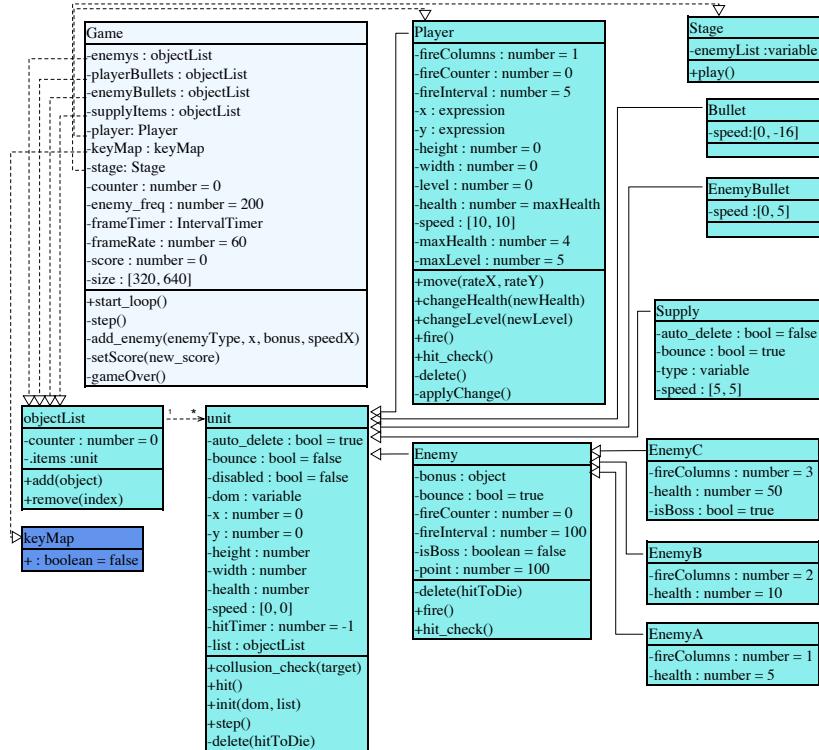
3.3 画面遷移



4 内部仕様

4.1 クラス図

このクラス図はチームメンバーと共に作ったものである。



4.2 各クラスの説明

- game :

ゲームの運行を制御し、全体モジュール（コントロールの入力、ゲームの表現、オブジェクトの管理、ステージの管理、スコアの計算）のベースとして総合クラスである。

- objectList :

unit の種類（enemy, supply, bullet etc.）により保存用配列として実装したオブジェクトリストクラスである。

- stage :

段階的に敵戦機の種類とか速度とか再生時間とか設計し再生関数を呼び出し用のマネージャークラスである。

- unit :

全ての独立単位オブジェクトのプロトタイプクラスである。幅と長さ、座標、スピードなど基本の属性初期化し、公用動作メソッドも定義する。

- player :

プレーヤー戦機の抽象クラスである。unit から継承し、その上に特有なライフ値と火力に関する属性を追加する

- enemy :

敵戦機の抽象クラスである。unit から継承し、その上に特有なライフ値と火力と戦機種類に関する属性を追加する

- enemyA : 普通の敵戦機のクラスである。

- enemyB : キャプテンの敵戦機のクラスである。

- enemyC : ボス敵戦機のクラスである。

- supply :

敵を倒れた時出る供給である。二つ種類で、ライフ値か火力の一つサポートする。

- bullet :

プレーヤー戦機から発射される弾である。

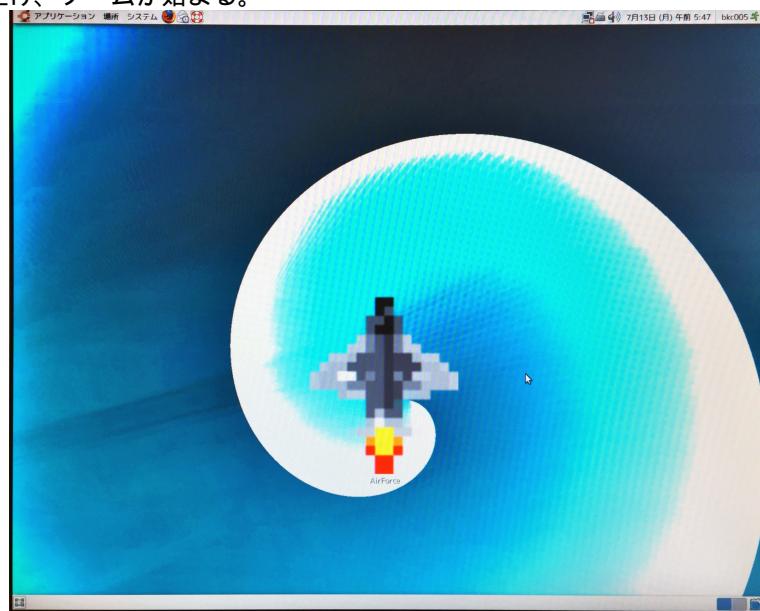
- enemyBullet :

敵戦機から発射される弾である。

5 実行例

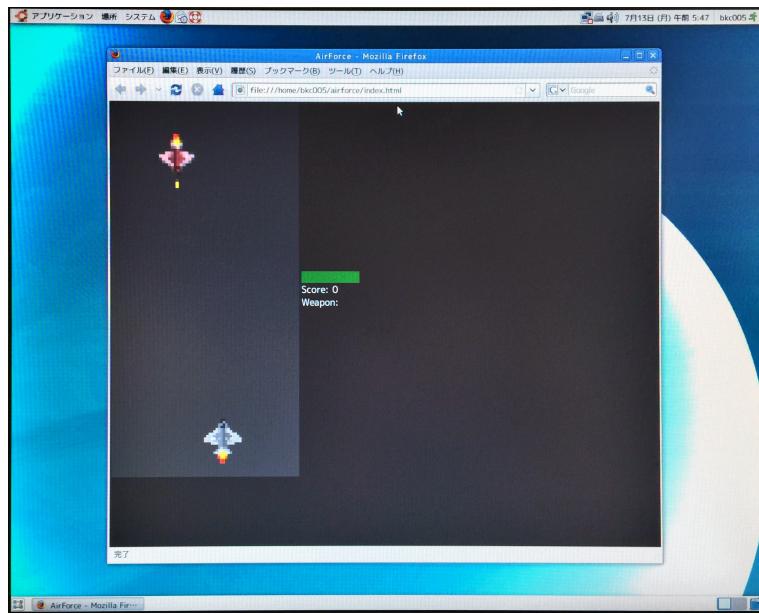
5.1 ディスクトップ

ディスクトップにゲームのアイコンをクリックすると、ブラウザが立ち上げ、ゲームが始まる。



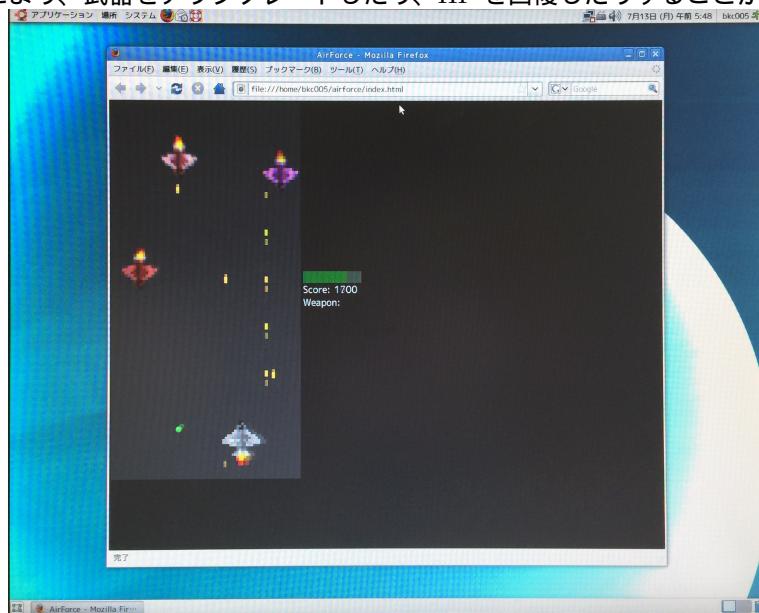
5.2 プログラム起動した直後

プログラムが起動したら、ゲームが自動的に始まる。キーボードでプレイヤーの戦機をコントロールすることができる。

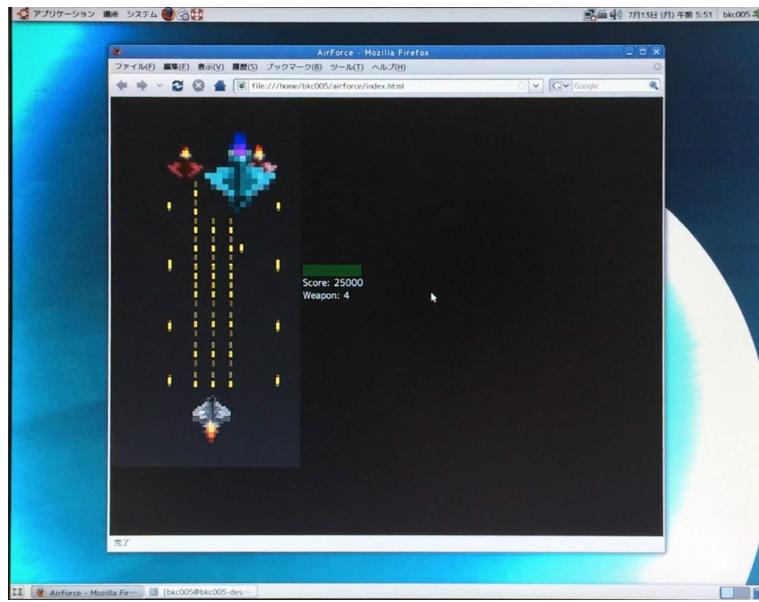


5.3 ゲーム画面

敵を倒せば、緑の円のサプライが出る場合がある。このサプライをとることにより、武器をアップグレードしたり、HPを回復したりすることができる。

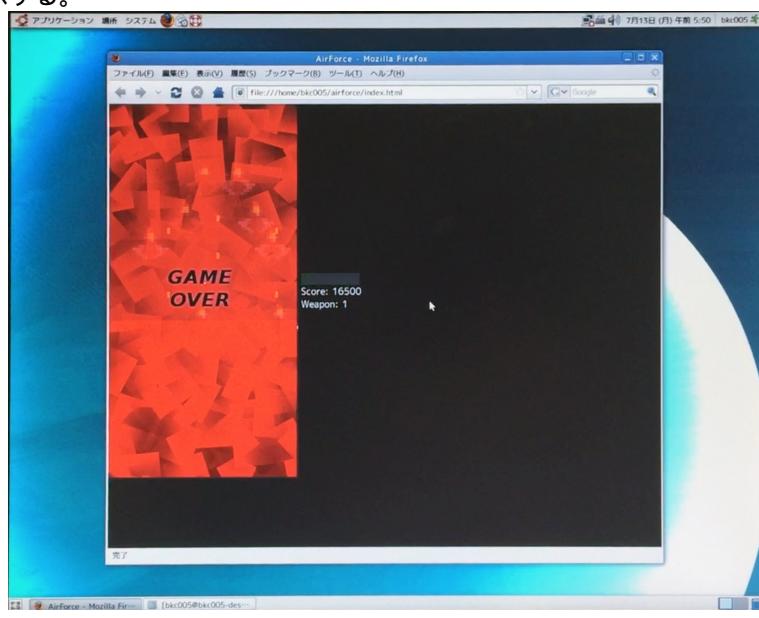


ゲームが進むと、敵が徐々強くなる。

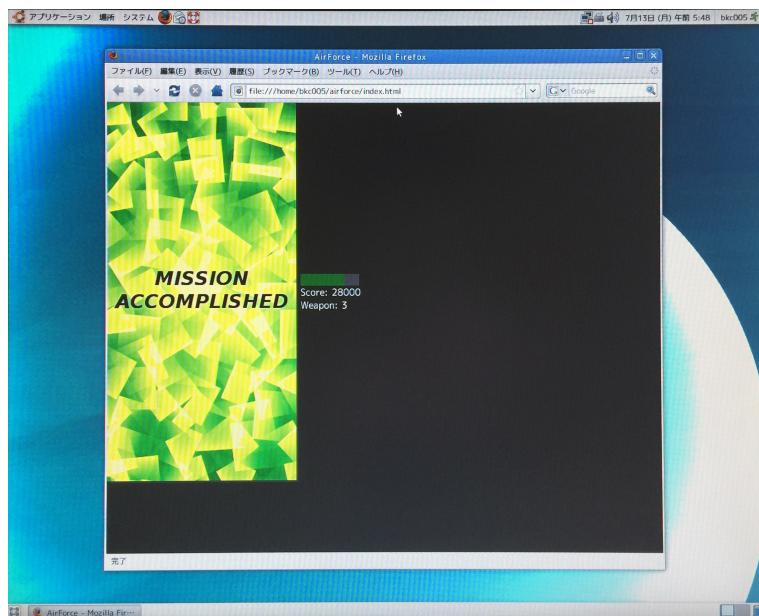


5.4 ゲームオーバー

プレーヤーが倒れば、ゲームオーバーとなり、ゲームオーバーの画面を表示する。



最終的ボスを倒せば、ゲームが終わり、勝利画面を表示する。



6 工夫した点

6.1 プログラム上

- JavaScript のプロトタイプチェンによるオブジェクト指向プログラミングで実装することにより、テストの便利性も考え、プログラムの各モジュールを独立させた。
- 共有するメソッドをプロトタイププロパティーにアサインすることにより、無駄なメモリ使用を削除し、良い性能を実現できた。
- オブジェクトの共通部分を継承ツリーとして作成したため、将来新しい敵、武器などを拡張しようとすると簡単にできる。

6.2 デザイン上

- GIMP を使いオリジナルな 8 bit スタイル画像素材を幾つか作成した。
- ゲームプレーの体験を改善する止めに、難しさをよく考え、段階的なシーンを設計する
- プレーヤーの武器の 5 段階のレベルを用意した。
- 敵を 3 種類を作り、それぞれ異なるヘルス、武器、速度などを設定した。

6.3 チームワーク

- GitHub を利用し、うまくチーム単位でのプログラミングができた。一方マルチスレッドの開発もできる。
- 開発中、ピアレビューしてエラーを発見し、ソフトのロバスト性を改善する。