Software Requirement Specifications for

Battle Ships

Version 1

18 October 2014

Index

Introduction

This Software Requirements Specification gives a whole idea of all the functions and features of the Battle ships game created by UG students of IIT Bombay (2014 autumn sem) as a part of CS 101 course project.

The name of the game created is Battle Ships. In the game, players have to fire bombs at each other's ships in the game till all the ships of one of the players is reduced to ashes i.e dead, when the game ends. The game is only available in single player. The game was made in C++ using GCC compiler in Code::Blocks IDE.

Functional Specifications

Functions of the Computer: The basic tasks performed by the system may be divided as follows:

In this the user is required to give input with which the computer marks it on the maze, the main task undertaken by the computer is using logic and guessing where to hit next ,and validating user input. The system is supposed play like a person would. The computer generates maze for itself and the player.

The various functions used throughout the program and a brief description:

1. void mazegeneraterplayer(int a[10][10]) : this function is used to generate a maze for the player.

2. void mazegeneratercomputer(int a[10][10]) : This function is used to generate a maze for the computer.

3. void display (int a[10][10],int h[10][10]) : This function is used to display the matrices and color code them accordingly.

4. int checkend(int h[10][10]) : This is used to check whether the game has finished or not.

5. int compchance(int b[10][10],int h[10][10],int xy[10][10],int nexthit[2],int l[2],int r,int g) : This is used to let the computer play and hit a block which is decided by the logic encoded in the function.
6. inline void mySleep(clock_t sec) : Used for inducing a time delay.
7. void displaym(int a[10][10],int h[10][10],int idm) : used to display the matrix in multiplayer version.

Description of Data

Description of Variables :

The program mainly required two Mazes one for each player and computer. Another maze was also used “hit” which was used to store where all the hits were done and where all ships died.

The types of data used can be broadly classified as :

1. Arrays

```
int
player[10][10]={0};              // showing player's ships
int
computer[10][10]={0};            // showing computer's ships
int hit[10][10]={0};           // showing hits on ships
                               // to get the last direction in a
int xy[10][10]={0};            cell
int l[2]={0,0};               // to pass the last location
int h[2]={0};               // to get the next hit position
int r=1;                   // tells to randomize or not
```

The player and the computer arrays are used to store the position of the ships . the hit array basically tells where previously bombs have been placed and where it was a miss and where the ships died.

the xy[][] tells the direction of hitting the computer has to go and is a vital part of the computer logic.

the array l[] tells the last place where the computer successfully bombed the players ship. h[] is used to get the position of next hit. The variable r specifies whether or not the computer should randomize its selection.

Description of libraries used :

The various libraries used in the program are :

Source defined:

```
#include<direct.h>
#include<iostream>
#include <cstdlib>
#include<time.h>
#include <stdio.h>
#include <stdlib.h>
#include "libs/rlutil.h"
#include <istream>
#include <fstream>
#include <iterator>
#include <ctime>
```

User Interface Requirements

Its mainly based on the terminal window and a basic C++ compiler and executor which follows C11 standards will do fine .

Acceptance Criteria

The criteria of acceptance is in project outlay which includes

- Basic gameplay – Maze generation and Plotting of Maze
- The number of ships damaged
- End game Criteria

- AI for single player
- Game menu
- Instructions screen

Appendices

Credits

CS101 Course Professor :

Prof. D.B.Phatak & Prof. Supratik Chakraborty

Lab Batch TA : Sahil

Batch Members

1. Franklin Varghese (Team Leader)
2. Sthitapragyan mohanty
3. Praveen chauhan
4. Anamika aich

Special Thanks

We would take this opportunity to thank

ARUN VINAYAK U (Team Leader) DOPPALAPUDI CHANDRAKANTH MULA REVANT BABU ANIL KUMAR JAYANTH (Team Leader)

for their wonderful project on Battle tanks which was a great help to us for gaining the format of the project.

We have used code from codeexchange.com for implementing color change of text, cplusplus.com, cppreference.com, Various websites for ideas related to game.