



# RELATÓRIO DO TRABALHO FINAL DE PROGRAMAÇÃO

CEFET/RJ - Campus Nova Iguaçu  
Alef de Pontes Silva  
Erickson Siqueira da Silva

Nova Iguaçu,  
Maio de 2019

# Conteúdo

<b>1</b>	<b>Introdução</b>	<b>2</b>
<b>2</b>	<b>Funções implementadas</b>	<b>2</b>
<b>3</b>	<b>Como compilar e rodar o programa</b>	<b>3</b>
<b>4</b>	<b>Como jogar</b>	<b>3</b>
<b>5</b>	<b>Dificuldades enfrentadas</b>	<b>6</b>
<b>6</b>	<b>Conclusão</b>	<b>7</b>

# Lista de Figuras

1	Menu principal . . . . .	3
2	Tela de jogo . . . . .	4
3	EE-T1 Osório . . . . .	4
4	M1-Abrams . . . . .	4
5	Tijolo . . . . .	4
6	Parede indestrutível . . . . .	4
7	Parede da borda . . . . .	5
8	Base . . . . .	5
9	Tanque do jogador . . . . .	5
10	Tanque virado para cima . . . . .	5
11	Tanque virado para baixo . . . . .	5
12	Tanque virado para a direita . . . . .	5
13	Tanque virado para a esquerda . . . . .	5
14	Representações dos tanques inimigos . . . . .	6
15	Lista de jogadores . . . . .	6

# 1 Introdução

Battle City é um jogo de tanques de guerra no qual joga, apenas, uma pessoa por vez. O objetivo do jogo é destruir todos os tanques inimigos para conseguir avançar nas fases. O jogador perde caso sua base seja destruída ou perca todas as suas vidas. Nesta adaptação, o jogo consiste em um jogo de tabuleiro, de turnos, com uma ação por turno (movimentar, atirar ou girar).

# 2 Funções implementadas

O jogo, atualmente, possui 2 modos: um normal e uma versão DEBUG. No modo normal, temos um tabuleiro 13x13, com inimigos e paredes geradas aleatoriamente, e a movimentação do tanque do jogador. O modo DEBUG é parecido com o primeiro modo; porém, ele informa todas as ações de movimento feitas pelo jogador.

Para a construção do jogo foram utilizadas as funções if else, switch case, while, do while, for, matriz e struct.

Os if's else's foram utilizados principalmente para as limitações do jogo, tanto de movimentação do jogador quanto de posicionamento dos objetos.

O switch case foi utilizado para a implementação de menu e suas respectivas escolhas.

O while e o do while mantém o jogo rodando até que o jogador escolha uma das opções de saída do jogo.

For foi uma função muito utilizada, ela é muito importante para preencher as matrizes e plotá-las na tela.

As matrizes são a função base do jogo, é dentro dela que tudo acontece, já que ela é a função que forma o campo de batalha.

O struct foi utilizado para armazenar as informações principais de cada tanque, como por exemplo a sua forma de representação na tela do jogo.

Utilizamos funções para uma melhor organização do programa, as funções foram de extrema ajuda para que a função principal do programa não ficasse extremamente grande.

Utilizamos arquivos para passar o mapa do para dentro do jogo. Também foi utilizada para a geração de uma lista de jogadores.

E strings para armazenamento de nome do jogador.

Os inimigos possuem uma inteligencia artificial,esta inteligencia foi implementada com uma sequencia de contadores de linha e coluna que calculam qual a menor distancia para conseguir atirar no tanque principal, perseguindo assim o tanque principal para aniquila-lo.

### 3 Como compilar e rodar o programa

Apos descompactar a pasta do jogo,abra o arquivo TBP.dev, compile e execute o programa. Utilizando o arquivo executável do jogo, dê um clique duplo sobre o arquivo TBP.exe para rodá-lo.

### 4 Como jogar

Ao abrir o jogo, será mostrado um menu inicial na tela. O menu possui 4 opções: Jogar, Jogar na versão DEBUG,Lista de ganhadores , Sair e uma breve apresentação dos tanques do jogo.

```
Escolha uma das opcoes:
[1]:Jogar;
[2]:Jogar no modo DEBUG;
[3]:Lista de ganhadores;
[4]:Sair.

V = EE-T1 Osorio;
T = M1-Abrams;
A = Jogador principal.

Opcao:
```

Figura 1: Menu principal

Ao seleccionar a opção Jogar ou Jogar no modo DEBUG, aparecerá na tela um campo 13x13, com inimigos, paredes a base e o tanque do jogador. Também aparecerá um breve menu de movimentação com instruções para jogar junto com seus pontos de vida e pontos.

O jogador perde quando as vidas chegarem a 0 ou a base for destruída, e o ganha caso conclua as 2 fases derrotando 3x quantidade de inimigos iniciais na fase.

Cada inimigo derrotado adiciona um valor de 10 pontos ao placar do jogador. Sempre que o jogador avança de nível seus pontos são multiplicados pela quantidade de vida.

Caso o jogador vença ele terá direito de adicionar seu nome na lista de ganhadores, essa lista contém o nome do jogador e sua pontuação.



Figura 2: Tela de jogo



Figura 3: EE-T1 Osório



Figura 4: M1-Abrams



Figura 5: Tijolo



Figura 6: Parede indestrutível



Figura 7: Parede da borda



Figura 8: Base



Figura 9: Tanque do jogador

No menu de movimentação, há as seguintes opções: "W" para movimentar na para cima; "S", para baixo; "D", para a direita; "A", para a esquerda,"G" para girar o tanque e "F", para atirar. A opção "F", que permite o jogador atirar na direção que o tanque esta apontado, dispara um projétil que destrói paredes e inimigos; porém, não destrói paredes indestrutíveis.

A opção girar implementa a opção de virar o tanque pra 4 direções, cima, baixo, esquerda e direita. Sempre que uma das direções é escolhida a forma de representação do tanque muda.



Figura 10: Tanque virado para cima



Figura 11: Tanque virado para baixo



Figura 12: Tanque virado para a direita



Figura 13: Tanque virado para a esquerda

Os inimigos possuem a capacidade de girar seus tanques e também atirar, assim como o jogador principal. É necessário muito cuidado por que eles possuem uma inteligência artificial.

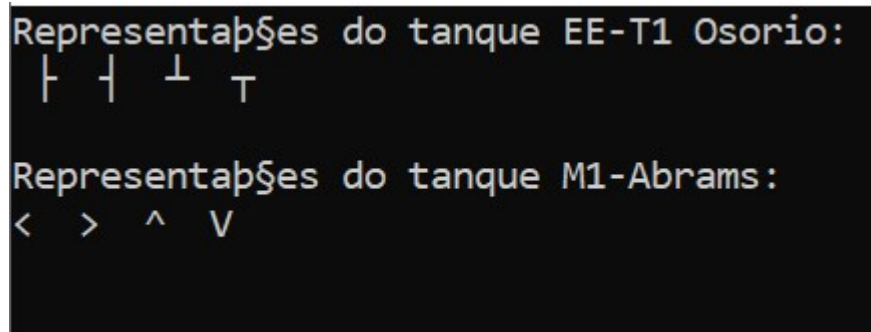


Figura 14: Representações dos tanques inimigos

A opção 3 representa uma lista de jogadores que já ganharam o jogo, nela o jogador pode conferir o nome de outros jogadores que já ganharam e a pontuação que obtiveram.

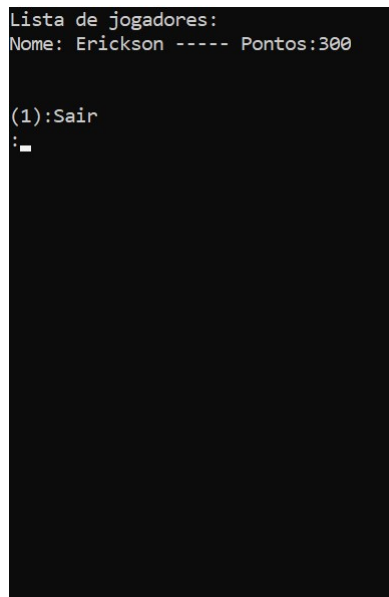


Figura 15: Lista de jogadores

## 5 Dificuldades enfrentadas

Realmente, essa segunda parte do trabalho não foi nada fácil. O programa teve de ser, praticamente, refeito! Acrescentar funções, implementar outras coisas pedidas e ter que mudar algo já feito porque, quando essas outras coisas foram acrescentadas, as já feitas pararam de funcionar, foi bastante comum em nosso trabalho.

Nessa parte, já não éramos mais três, porém, dois. Um membro do grupo não pôde mais estar presente por ter que se dedicar quase que integralmente ao CPOR.

A maior dificuldade foi, sem dúvidas, a de que apenas um de nós tem um computador; o outro usa um aplicativo de celular. Por muitas coisas novas terem sido acrescentadas, o programa do jogo já não funcionava mais no aplicativo, o que impossibilitou o que possui somente o celular de trabalhar no jogo em casa. Por tal motivo, foi necessário um enorme esforço do outro membro, e também foi necessário irmos nas monitorias com um notebook emprestado, pois somente um computador da sala de monitoria tem o Dev C++ atualizado o bastante para trabalharmos bem, e ele nem sempre funcionava ou estava livre.

## 6 Conclusão

Esse trabalho foi de grande auxílio a nossa aprendizagem; afinal, foi necessário saber diversos comando para poder conseguir trabalhar com um programa de mais de mil linhas. Também foi necessário melhorar o uso de nosso tempo; muitas horas foram gastas em cima do jogo. E o fato de um programar pelo computador e o outro, pelo celular, também exigiu criatividade e sacrifícios pessoais para que ambos trabalhassem em harmonia pelo jogo.

Tivemos que ser pacientes e cooperadores para fazer o jogo, o que tornou o grupo mais unido. Damos nosso melhor no jogo e ficamos feliz com o resultado.

Apesar de um de nós trabalhar pelo celular, o outro, pelo computador, e ou outro, estar bastante ausente, conseguimos nos harmonizar muito bem. Tais dificuldades só serviram pra nos ajudar a melhorar e procurarmos criativas soluções para enfrentá-las.