

## **Documentação do Jogo**

# Space Maths

Autores:

André Teiichi Santos Hyodo

Belo Horizonte

Julho de 2023

## Documentação do Sistema

*O Space Invaders é para o desenvolvimento de um jogo na linguagem C++ e com as bibliotecas do Allegro5.*

*O jogo se baseou no clássico jogo Space Invaders e com alterações no ambiente para que se tornasse educativo. Com isso, o jogo se voltou para o público infanto-juvenil que já tem conhecimentos básicos sobre matemática.*

*O jogador terá em mãos o controle da nave e dos seus tiros com o teclado que devem ser direcionados aos blocos com possíveis respostas do problema matemático proposto na tela. A partir do momento que o usuário acerta a resposta, seus pontos aumentam e, ao errar ou colidir a nave com o cometa, perderá uma vida, que ao ser zerada, finalizará o jogo.*

Documentação do Software .....	2
Introdução.....	4
Escopo do software .....	4
Nome do sistema e de seus componentes principais .....	4
Missão ou objetivo do software .....	<b>Erro! Indicador não definido.</b>
Arquitetura do Sistema.....	7
Funcionalidades do produto .....	7
Usuários e sistemas externos .....	8
Descrição .....	8
Documentação do código.....	9
Documentação da Estrutura de dados geral do software .....	9
Função <nome da função> .....	10
Função <nome da função> .....	10
Testes do software.....	11
Casos de testes do software: função <nome da função> .....	11
Casos de testes do software: função <nome da função> ..	<b>Erro! Indicador não definido.</b>

## ***Introdução***

### **Escopo do Jogo**

#### ***Nome do jogo e de seus componentes principais***

*Space Maths*

#### ***Análise do público-alvo***

*O Space Maths é voltado para o público infantil/adolescente.*

#### ***Pré-requisitos***

*Conhecimentos prévios de operações matemáticas básicas (soma, subtração e multiplicação).*

#### ***Objetivos educacionais e específicos do jogo***

*O Space Maths tem como objetivo desenvolver os conhecimentos matemáticos, mais especificamente os comandos de soma, subtração e multiplicação.*

### **Itens de Avaliação**

*Questões Matemáticas: O jogo apresenta diversas operações matemáticas, podendo avaliar o jogador com base nos seus conhecimentos e capacidade de raciocínio rápido.*

*Pontuação: Um sistema de pontuação será implementado para avaliar o desempenho dos jogadores. Os jogadores receberão pontos a cada acerto nas operações realizadas.*

*Níveis de Dificuldade: O jogo apresentará um nível de dificuldade baseado na velocidade do jogo e, consequentemente, na velocidade de raciocínio necessário para realizar as operações matemáticas.*

*Feedback do Jogo: O jogo fornecerá feedback imediato sobre as respostas dos jogadores. Quando uma resposta estiver correta, o jogador receberá um ponto e verá o cometa sendo destruído. Caso a resposta esteja incorreta, o perderá uma vida, podendo perder até 2 sem que perca o jogo.*

*Estatísticas de Desempenho: O jogo registrará os pontos feitos pelo jogador, que serão referentes ao número de acertos feitos.*

### **Estratégia de apresentação**

*Introdução Temática: O jogo será ambientado no espaço, baseado no clássico jogo Space Invaders, trazendo uma jogabilidade similar e adicionando funcionalidades para tornar o jogo educacional e auxiliar no desenvolvimento intelectual dos jogadores.*

*Design de Interface Intuitiva: A interface do jogo será projetada de forma clara e intuitiva. Os elementos visuais, como a nave espacial do jogador, os cometas e as operações matemáticas, serão distintos e facilmente identificáveis. Isso permitirá que os jogadores compreendam rapidamente os elementos do jogo e se concentrem nas operações matemáticas sem distrações.*

*Graduação de Dificuldade: O jogo apresentará uma progressão gradual de dificuldade à medida que os jogadores avançam no jogo. No início, a velocidade com que o jogador precisará realizar as operações matemáticas é baixa, tendo bastante tempo para isso e, à medida que as operações são realizadas com sucesso, a velocidade aumentará gradualmente.*

*Variedade de Operações Matemáticas: O jogo apresentará uma variedade de operações matemáticas, incluindo adição, subtração e multiplicação e divisão. Essa variedade garantirá que os jogadores tenham a oportunidade de praticar diferentes tipos de operações e fortalecer suas habilidades matemáticas de maneira abrangente.*

*Feedback Visual e Sonoro: O jogo fornecerá feedback visual e sonoro imediato para cada resposta do jogador. Quando uma resposta estiver correta, o jogador ouvirá um som de sucesso. No caso de uma resposta incorreta, o jogador receberá um feedback visual e sonoro indicando o erro. Esse feedback ajudará os jogadores a identificar suas respostas corretas e incorretas.*

## Arquitetura do Sistema



## Funcionalidades do produto

Número	Funcionalidade do sistema
1	Botão para inicialização do jogo
2	Nave controlada pelo usuário
3	Cometas como Alvos
4	Operações matemáticas para ser resolvidas
5	Controle de tiros da nave pelo jogador
6	Tabela de pontuação
7	Tabela de “vidas” restantes
8	Tela de finalização com pontuação final
9	Botão para reiniciar o jogo após a sua finalização

## Usuários e sistemas externos

### *Descrição*

<i>Número</i>	<i>Usuários</i>	<i>Definição</i>
<i>1</i>	<i>Estudantes</i>	<i>Os estudantes são os principais usuários do jogo Space Maths.</i>
<i>2</i>	<i>Professores</i>	<i>Pode utilizar o Space Maths para auxiliar no ensino matemático para seus alunos.</i>
<i>3</i>	<i>Fóruns e Comunidades Online</i>	<i>O jogo pode oferecer fóruns ou comunidades online dedicadas, onde os jogadores podem interagir, compartilhar dicas, solicitar suporte ou fornecer feedback sobre o jogo.</i>



## **Documentação do código**

### ***Documentação da Estrutura de dados geral do software***

*Structs: usados para guardar os dados de dados do tipo NaveEspacial, Projeteis, Estrelas, Calculo e Mouse ;*

*Funções de áudio do alegro, do tipo ALLEGRO\_SAMPLE, ALLEGRO\_SAMPLE\_INSTANCE;*

*Objeto: TECLAS para armazenar as variáveis que associam as teclas de cima, baixo, direita, esquerda, space e enter;*

*Funções para fonte de letra: ALLEGRO\_FONT para ajustar o tipo de fonte e tamanho da letra.*

*Funções para naves: InitNave, DesenhaNave;*

*Funções para movimentação: MoveNaveCima, MoveNaveBaixo, MoveNaveDireita, MoveNaveEsquerda;*

*Funções para tiros: InitBalas, AtiraBalas, AtualizarBalas, DesenhaBalas, BalaColidida;*

*Funções para cometas: InitCometas LiberaCometas, AtualizarCometas, DesenhaCometas, CometaColidido;*

*Funções para plano de fundo: InitPlanoFundo, AtualizarPlanoFundo, DesenhaPlanoFundo;*

*Eventos: ALLEGRO\_EVENT\_DISPLAY\_CLOSE, ALLEGRO\_EVENT\_KEY\_DOWN, ALLEGRO\_EVENT\_KEY\_UP, ALLEGRO\_EVENT\_MOUSE\_AXES, ALLEGRO\_EVENT\_MOUSE\_BUTTON\_DOWN, ALLEGRO\_EVENT\_TIMER;*

### **Funções**

```
void InitNave (NaveEspacial &nave);  
void DesenhaNave(NaveEspacial &nave);  
void MoveNaveCima(NaveEspacial &nave);  
void MoveNaveBaixo(NaveEspacial &nave);  
void MoveNaveDireita(NaveEspacial &nave);  
void MoveNaveEsquerda(NaveEspacial &nave);  
void InitBalas(Projeteis balas[],int tamanho);  
void AtiraBalas(Projeteis balas[], int tamanho, NaveEspacial nave);  
void AtualizarBalas(Projeteis balas[], int tamanho);  
void DesenhaBalas(Projeteis balas[], int tamanho);  
void BalaColidida(Projeteis balas[], int b_tamanho, Cometas cometas[], int  
c_tamanho, NaveEspacial &nave, Calculo &calc);  
void InitCometas(Cometas cometas[], int tamanho, Calculo calc);  
void LiberaCometas(Cometas cometas[], int tamanho);  
void AtualizarCometas(Cometas cometas[], int tamanho);  
void DesenhaCometas(Cometas cometas[], int tamanho, ALLEGRO_FONT *font25,  
Calculo calc);  
void CometaColidido(Cometas cometas[],int c_tamanho, NaveEspacial &nave);  
void InitPlanoFundo(Estrelas estrelas_pf[][NUM_ESTRELAS],int pf_tamanho, int  
e_tamanho);  
void AtualizarPlanoFundo(Estrelas estrelas_pf[][NUM_ESTRELAS],int pf_tamanho, int  
e_tamanho);  
void DesenhaPlanoFundo(Estrelas estrelas_pf[][NUM_ESTRELAS],int pf_tamanho, int  
e_tamanho);  
void RetornaCalc(Calculo &calc);  
void Options(int tamanho,Calculo &calc);  
void EscreveOptions(Cometas cometas[], int tamanho, ALLEGRO_FONT *font14,  
Calculo calc);
```

## Testes do sistema

### Casos de testes do sistema: função BalaColidida

Número	Varáveis de Entrada	Valores válidos	Resultado Esperado	Valores inválidos	Resultado Esperado
1	Balas[]	Ativo, x , y	X>0 e x<largura/t X>0 e x<altura/t	X<0 e x>largura/t X<0 e x>altura/t	
2	b_tamanho	Numero de balas a serem disparadas	b_tamanho>0	b_tamanho<0	
3	cometas[]	Ativo, x , y	X>0 e x<largura/t X>0 e x<altura/t	X<0 e x>largura/t X<0 e x>altura/t	
4	c_tamanho	Número de cometas	c_tamanho>0	c_tamanho<0	
5	nave	Pontos e vidas	Pontos>0 Vidas>0	Pontos<0 Vidas<0	