

## APRESENTAÇÃO

Você já ouviu falar em *Space Impact*?

*Space Impact* é um jogo do gênero *shoot 'em up* que fez bastante sucesso em seus primórdios. Suas primeiras versões eram muito comuns em celulares antigos da Nokia e foram extremamente populares no Brasil no início dos anos 2000.

Diferente de seu, podemos dizer, "predecessor", *Space Invaders*, *Space Impact* trazia uma dinâmica muito mais intensa, com ondas de inimigos mais longas e diversificadas, além de chefes colossais com mecânicas de batalha bem interessantes.

O jogo não ficou restrito à sua versão inicial e passou por várias evoluções e lançamentos ao longo dos anos:

- *Space Impact*
- *Space Impact: Meteor Shield*
- *Space Impact: Kappa Base*
- *Space Impact Light*
- *Space Impact 2*
- *Space Impact Evolution*
- *Space Impact Evolution X*
- *Space Impact 303*
- ...

Neste trabalho, vamos implementar uma versão alternativa do *Space Impact*, particularmente baseada no *Space Impact 303* (com as mesmas dinâmicas) — você pode conferir um *gameplay* desse jogo no [YouTube](#). Para isso, haverá uma série de requisitos mínimos que representarão 80% da sua nota. Além disso, você poderá propor e implementar funcionalidades extras, que complementarão os outros 20% — para que os

20% adicionais sejam avaliados, no entanto, você deve garantir, pelo menos, 50/80 dos requisitos mínimos.

A seguir, descreveremos em detalhes as funcionalidades esperadas para o jogo e as fases e de desenvolvimento.

## **REQUISITOS MÍNIMOS**

- O jogo deve conter, pelo menos, DUAS FASES, cada uma com um CENÁRIO DIFERENTE;
- A NAVE DO JOGADOR do jogo deve ter SPRITE, e uma pequenas ANIMAÇÕES relacionadas à movimentação para esquerda e direita;
  - A nave deve ter um sistema de HP (vida), que quando zerado, indica fim de jogo;
  - A nave deve ter um ataque simples sempre disponível;
  - Veja, adiante, os comentários sobre ataques especiais.
- Em CADA FASE deve existir, pelo menos, DOIS tipos de INIMIGOS normais (diferentes entre si, no mínimo, quatro tipos ao total);
  - Dois dos quatro inimigos devem LANÇAR PROJÉTEIS;
  - Apenas um dos quatro inimigos pode ser imóvel, os outros devem ter algum PADRÃO DE MOVIMENTAÇÃO (mesmo que simples);
  - Cada tipo de inimigo deve ter uma *SPRITE* ÚNICA, sem repetição.
- Cada fase deve ter uma DURAÇÃO mínima de TRINTA SEGUNDOS;
- Cada fase deve ter um CHEFE (inimigo único - dois ao total, no mínimo) no final dela;
  - Cada chefe deve ter uma *SPRITE* ÚNICA;
  - Cada chefe deve ter, no mínimo, dois ATAQUES ÚNICOS, sendo, no mínimo, um deles relacionado ao LANÇAMENTO DE PROJÉTEIS;
  - Os chefes devem ser BEM MAIS RESISTENTES que os inimigos normais;
  - (Subjetivo) O chefe deve apresentar algum DESAFIO ao jogador – por exemplo, a depender da sua vida, aumentar a dificuldade do mesmo.
- Devem existir, pelo menos, dois ATAQUES ESPECIAIS para a NAVE DO JOGADOR.
  - Os ataques devem ser ADQUIRIDOS NO MAPA;
  - Os ataques especiais devem ser TEMPORÁRIOS (no máximo, cinco segundos cada).

# **ETAPAS DE TRABALHO**

## **Etapa 1: Projeto**

Todos os alunos devem entregar um documento em PDF, organizado por seções, descrevendo o projeto inicial de implementação.

Esse documento deve conter as seguintes informações: segmentação do trabalho (quais arquivos e módulos você pretende criar e por quê), dependências entre os módulos a serem desenvolvidos, a ordem de desenvolvimento desses módulos e, por fim, um cronograma estimado para o desenvolvimento do programa. O **PDF do projeto deverá ser entregue até o dia 05/11**, no Moodle.

**A NÃO ENTREGA DO DOCUMENTO RESULTARÁ EM NOTA ZERO NO SEGUNDO TRABALHO E AVALIAÇÃO.**

## **Etapa 2: Desenvolvimento**

O **período para o desenvolvimento do trabalho será de 24/10/2024 até 08/12/2024, às 23:59.**

Durante esse período, você deve realizar a implementação e preparar o trabalho para a entrega. Em todas as aulas presenciais haverá espaço para tirar dúvidas, então evitem ficar bloqueados em algum ponto da implementação.

O trabalho deve ser implementado em C, utilizando a biblioteca Allegro 5 – a entrega será no Moodle.

## **Etapa 3: Defesas e Modificações**

O **período para as defesas dos trabalhos será de 09/12/2024 a 11/12/2024.** Horários extras serão disponibilizados para garantir que todos tenham tempo para apresentar.

Cada aluno terá cerca de 15 minutos para defender e demonstrar as funcionalidades implementadas no trabalho, seguidos de 30 minutos para realizar uma pequena modificação no código, que será **muito simples** e definida especificamente para cada trabalho no momento da defesa.

Todos devem agendar seus horários de apresentação, considerando a necessidade de até 45 minutos consecutivos.

## (SUGESTÃO) HISTÓRIA

Para a nota complementar (referente aos 20%), você **pode optar por desenvolver um enredo** para o jogo, o que valerá metade da pontuação adicional (ou seja, 10/20).

Para isso, será necessário incluir no seu jogo os seguintes elementos:

1. Uma tela inicial no estilo "*Star Wars*", apresentando o contexto do jogo e a missão geral do protagonista;
2. Uma fala do protagonista no início da primeira fase, indicando qual é sua missão naquela fase;
3. Uma fala do chefe ao final da primeira fase, desafiando o protagonista ou indicando sua derrota;
4. Os mesmos requisitos apresentados nos itens 2 e 3 devem ser aplicados à segunda fase;
5. Uma tela final, também no estilo "*Star Wars*", apresentando o resultado da missão do protagonista após ser cumprida e as consequências no contexto do jogo.

## DICAS

- Allegro 5: <https://liballeg.org/>
- Documentação do Allegro 5: <https://liballeg.org/a5docs/trunk/>
- Boa fonte de *sprites*: <https://itch.io/game-assets/free/tag-spaceship>

## NOTAS IMPORTANTES

- Você deve preparar um ***makefile*** (pelo menos, com operações *make* e *make clean*) para o programa;
- ***Segmentation faults*** não são aceitáveis;
- A entrega deve ser feita por meio de um **arquivo .zip**, este contendo todos os arquivos requisitados e necessários para a compilação e execução do programa desenvolvido. O nome do arquivo deve ser **APENAS o seu login** nos sistemas do

DINF. Por exemplo, se o meu login é vinicius, o nome do arquivo deve ser **vinicius.zip**;

- Não cumprir os critérios de entrega acarretará um desconto de 5% da nota máxima prevista (o trabalho valerá 95% do peso ao invés de 100%).
- Eventuais **anexos** poderão ser incluídos neste documento para **esclarecer dúvidas e persistir acordos** feitos em sala de aula.
  - Todas as modificações ou especificações extras serão acordadas em aula e indicadas em atualizações deste documento; não comparecer às aulas ou não consultar o documento periodicamente são decisões arriscadas;
  - Na dúvida de algo, pergunte antes de fazer – seja na aula ou por e-mail;
  - Se você decidir entregar o trabalho antes do prazo, o mantenha como rascunho no sistema – para que você possa fazer eventuais modificações e reentregas.