

#### Universidade do Vale do Itajaí Escola Politécnica Curso de Ciência da Computação

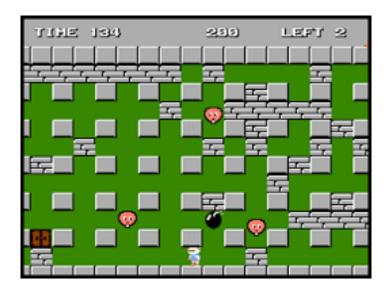
#### Algoritmos e Programação II (22817)

### Definição do trabalho da M2 - BOMBERMAN

**Data da defesa:** No Material

**Modalidade:** em DUPLAS ou em TRIO. (trabalhos individuais perderão 20% da nota)

Visão Geral:



Bomberman é uma série de jogos de estratégia, inicialmente desenvolvido pela fabricante Hudson Soft. O jogo original foi lançado em 1983 e novos jogos da série ainda continuam a ser lançados. Hoje, Bomberman estreia em mais de 60 jogos diferentes. (Wikipedia, 2023)

## COMPOSIÇÃO DO JOGO

Atender a todos os critérios do trabalho anterior, caso não seja atendido, terá desconto, conforme itens anteriores.

- O Jogo deve estar otimizado (pontuação 0.4)
- O Jogo não deve piscar ou deslocar blocos sem a que seja da dinâmica do Jogo (pontuação 0.4)
- Design Caprichado (pontuação 0.4)
- O Jogo deve ter Menu (pontuação 0.4)
  - Ter opção explicando como funciona o Jogo
  - Ter opção de Dificuldades e explicando os itens
- Ter opção explicando a pontuação (Cálculo deve ser implementado, levando em consideração, movimento, bombas utilizadas, inimigos abatidos, caixas destruídas....)
  - Ter opção mostrando o Rank Ordenado
- O Jogo deve permitir continuar/reiniciar quando o personagem morrer ou acabar (pontuação 0.4)
- O Jogo ser salvado em um arquivo (pontuação 0.4)

- Data, Jogador, Pontuação, Bombas, Movimentos ...
- O Jogo deve ter ao menos 3 fases (pontuação 0.4)
- No último nível deve ter 1 inimigo especial que execute movimentos mais inteligentes (pontuação 0.4)
- Os blocos quebráveis devem ser aleatórios (pontuação 0.4)
  - Deve-se garantir que o jogador tenha espaço para estourar as bombas
- Deve-se ter um portal para a próxima Fase (pontuação 0.4)
- O Jogo deve ter 3 dificuldades Ideia (pontuação 1.0)
  - Fácil movimento 3 inimigos mesmos movimentos do trabalho 1
  - Médio mínimo 5 inimigos, (ideia) com %50 de chance de se mover para o Jogador
  - Difícil mínimo 7 inimigos, (ideia) inimigos devem ter %75 de chance de se mover para o Jogador.
- Ter informações em Tela (pontuação 0.3)
  - Ter o tempo em Tela
  - Ter a pontuação em Tela, contagem de movimento, contagem da quantidade de bombas usadas
  - Ter itens pegos
- Inserir itens especiais, que deve ser exibido embaixo das caixas quebradas mínimo estes itens ex: (0.7)
  - Aumento do raio de fogo em 1 (cumulativo)
  - Aumento na quantidade de bombas (cumulativo)
  - Vida Extra (cumulativo)
  - Bomba Relógio
  - Permitir sobreviver a bomba
  - Permitir passar pelos blocos quebráveis
  - A explosão da bomba passará por tantos blocos quebráveis quanto o nível do fogo permitir.
  - \*\*\*Quando passar de fase os bônus restam\*\*\*
- O Jogo deve ter a opção de um segundo Jogador (1 Ponto extra na prova da M1)
- Colocar música no jogo (1 Ponto extra na prova da M1)

### Critérios técnicos (pontuação 4.0)

- Criar arquivo, manipular: ler, inserir e dados
- Parâmetro Default e Sobrecarga
- Struct e Template
- Recursividade

## <u>OBSERVAÇÃO</u>

Esse jogo tem por objetivo avaliar a utilização de técnicas de programação aprendidas na disciplina, principalmente quanto ao uso de sub-rotinas.

Além disso produzam todo o código pensando na possibilidade de que novas funcionalidades poderão ser solicitadas no futuro.

### **ENTREGA**:

• Postar um arquivo texto com o link para o trabalho no Guithub

#### **DEFESA**:

- O professor irá realizar a defesa do trabalho com o grupo, não havendo a necessidade de apresentar o trabalho para toda a turma.
- O professor poderá fazer perguntas sobre qualquer parte do código a qualquer um dos integrantes do grupo.
- Nota da apresentação será individual, além disso todas as notas relativas ao código dependem do desempenho na apresentação. Sem apresentação o trabalho terá nota ZERO.
- Para acelerar a apresentação dos trabalhos a mesma será feita em duas etapas:
  - Na primeira etapa da apresentação a equipe apresenta junto podendo ser feitas perguntas individuais. (a nota da apresentação ainda é individual)
  - Na segunda etapa da apresentação serão solicitadas modificações no código apresentado que deverão ser feitas de forma individual (sem auxílio dos outros integrantes da equipe).

# Critérios de Avaliação:

- 1. Funcionalidades = 60% da nota.
  - a. Só pontuará se souber explicar como foi implementado.
  - b. São 15 itens e cada item vale 0.4
- 2. Técnicas = 40% da nota.
  - a. Só pontuará se souber explicar como foi implementado.
  - b. São 4 itens e cada item vale 1.0 pontos

Obs1: Cópia de soluções de colegas ou da internet acarretarão nota 0 para todos os envolvidos. Obs2: Trabalhos entregues e/ou apresentados em atraso receberão um desconto inicial de 20% na nota final além de um desconto progressivo de 10% por semana de atraso. Ou seja, apresentar na semana seguinte a data estipulada deixa a nota máxima do trabalho em 70%.