



Visão geral do Projeto "Lógica em Jogo"

Coordenador:

Prof. Thiago Goveia

Colaboradores:

- Carlos Gabriel Pereira
- Gabriel Pereira Clóvis
- Kaio Siqueira da Silva
- Samuel Augusto

Carga-horária:

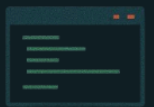
Início: 06/08/2025

Término: 14/01/2026

Carga horária total: 14 quartas-feiras

Resumo:

O projeto "lógica em jogo" tem como objetivo **estimular o raciocínio lógico e o pensamento computacional** em alunos do 2º e 3º ano do ensino médio do curso técnico de automação industrial e em alunos dos primeiros semestres do curso de Engenharia elétrica do IFMG, campus Itabirito. A utilização de jogos como ferramentas de aprendizagem tem se mostrado eficaz para **aumentar o engajamento** dos alunos e para **tornar conceitos abstratos mais palpáveis**, como é o caso da lógica de programação. Os jogos deste projeto: **pedra, papel e tesoura, forca, jogo da velha, e batalha naval**, foram **estrategicamente escolhidos** pois abordam conceitos-chave de programação em diferentes níveis. Em cada sessão, os alunos terão a oportunidade de **jogar entre si, para a identificação das regras** de cada jogo. Posteriormente, será feita a **modelagem** do jogo por meio de fluxogramas e a **implementação** em linguagem C, **em pares**. Após a implementação, **cada equipe testará o código de outras equipes**, pensando em casos de teste para fazer o programa falhar. Ao final do projeto, será feita uma **introdução aos sistemas web** e um dos jogos será traduzido para javascript e publicado na internet.



Organização!

Os jogos são o coração deste projeto! No entanto, para o máximo aproveitamento, antes de serem realizadas as atividades com os eles, será feita uma introdução ao controle de versão com Github e em cada sessão, uma revisão dos conceitos necessários de lógica de programação.

O Github será uma ferramenta fundamental, pois **resolve problemas como:**

- "Funcionava ontem, hoje não funciona mais!"
- "Perdi meu código!"
- "Não lembro o que mudei!"
- "Como trabalhar em dupla no mesmo código?"

Benefícios do GitHub:

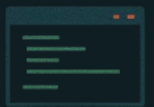
- Histórico completo de mudanças
- Backup na nuvem
- Colaboração em equipe
- Possibilidade de desfazer alterações

Os Jogos!

Cada jogo foi escolhido estrategicamente, por serem conhecidos, divertidos e por abordarem conceitos-chave de programação. São eles:

Pedra, Papel e Tesoura:

- **Geração de Números Aleatórios:** Para que o computador possa "escolher" entre pedra, papel ou tesoura.
- **Estruturas Condicionais (if/else if/else):** Determinar o vencedor com base nas regras do jogo (Pedra ganha de Tesoura, Tesoura ganha de Papel, Papel ganha de Pedra).
- **Entrada e Saída (Input/Output):** Receber a escolha do jogador e exibir o resultado da rodada.
- **Laços de Repetição (while):** Permitir múltiplas rodadas do jogo.



Forca:

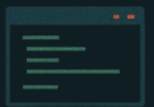
- **Strings e Manipulação de Texto:** Representar a palavra secreta e as letras adivinhadas.
- **Estruturas de Dados (Vetores):** Armazenar as letras já ditas.
- **Estruturas Condicionais (if/else):** Verificar se a letra adivinhada está na palavra, se o jogador ganhou ou perdeu.
- **Laços de Repetição (while):** Controlar o jogo enquanto a palavra não for totalmente adivinhada ou o número de erros for excedido.
- **Contadores:** Acompanhar o número de erros do jogador.

Jogo da Velha:

- **Estruturas de Dados (Matrizes):** Representar o tabuleiro do jogo e armazenar as jogadas de cada jogador (X ou O).
- **Estruturas Condicionais (if/else if/else):** Verificar as condições de vitória (linhas, colunas, diagonais) e empate.
- **Entrada e Saída (Input/Output):** Receber a jogada do usuário e exibir o estado atual do tabuleiro.
- **Laços de Repetição (for/while):** Iterar sobre o tabuleiro para verificar condições ou para controlar o fluxo do jogo até o final.
- **Funções:** Organizar o código em blocos reutilizáveis, como funções para desenhar o tabuleiro, verificar vitória, processar jogada, etc.

Batalha Naval:

- **Estruturas de Dados (Matrizes/Vetores):** Representar os tabuleiros (o seu e o do oponente) onde os navios estão posicionados e os tiros são marcados.
- **Estruturas Condicionais (if/else if/else):** Verificar se um tiro acertou um navio, se um navio foi afundado, ou se o jogador ganhou.
- **Laços de Repetição (for/while):** Controlar o fluxo do jogo em turnos e verificar o estado do tabuleiro.
- **Funções:** Para organizar a lógica do jogo, como posicionar navios, dar um tiro, verificar acertos e afundamentos.
- **Lógica de Coordenadas:** Manipulação de índices de matrizes para representar as posições no tabuleiro.



Cronograma:

As sessões do projeto envolvem a compreensão das regras dos jogos, sua modelagem, implementação e testes. Cada mês será destinado ao desenvolvimento de um jogo:

- **Agosto-Setembro:** Pedra, Papel e Tesoura
- **Outubro:** Jogo da Forca
- **Novembro:** Jogo da Velha
- **Dezembro:** Batalha Naval
- **Janeiro:** Programação Web

Os alunos jogarão entre si o jogo proposto em sua versão analógica para a identificação das regras e compreensão de suas dinâmicas. Em um segundo momento, além das regras padrão, cada um pode criar suas próprias regras personalizadas.

Em seguida será feita a modelagem do jogo utilizando fluxogramas e esquemas. Essa etapa visa desenvolver o pensamento computacional através da abstração e decomposição dos problemas dos jogos em etapas lógicas.

Uma vez desenvolvidos os modelos, será feita a implementação em pares, por meio da linguagem C. Após a codificação, cada equipe testará o jogo de outras equipes, pensando em casos de teste para fazer o programa falhar! Essa prática reforça a compreensão da lógica, estimula o pensamento crítico e a capacidade de identificar e solucionar problemas!

Publicação na internet:

Esta etapa final visa concretizar o aprendizado, proporcionando aos alunos a experiência de disponibilizar seu primeiro software na internet.

A publicação consistirá na tradução de um dos jogos para o ambiente web, por meio do uso das linguagens Javascript, HTML e CSS. A escolha destas tecnologias tem como motivação, apresentar aos alunos o processo de desenvolvimento de software mais próximo da realidade do mercado, isto é, a criação de aplicações para a internet.