

# **RELATÓRIO TÉCNICO – OTIMIZADOR GENÉTICO DE PERSONAGENS D&D;**

## **Integrantes:**

Cauan Teixeira Machado

João Felipe Quintino

## **1. Problema Proposto**

O problema consiste em otimizar a construção de personagens de RPG (Dungeons & Dragons) utilizando dados reais de personagens e monstros. O objetivo é maximizar o DPR (Damage Per Round) considerando atributos, classe, raça e arma escolhida, simulando desempenho contra diferentes valores de Classe de Armadura (AC) de monstros.

## **2. Modelagem do Problema**

Cada indivíduo do algoritmo genético representa um personagem codificado como um vetor: [classe, raça, arma, STR, DEX, CON, INT, WIS, CHA]. A função de aptidão calcula o dano médio esperado contra uma amostra de monstros, considerando modificadores de atributo, bônus de proficiência e probabilidade de acerto.

## **3. Algoritmo Genético Utilizado**

O algoritmo segue as etapas clássicas: inicialização, avaliação, seleção, reprodução, mutação e elitismo.

```
Iniciar população P com N indivíduos aleatórios
Para cada geração:
    Avaliar fitness de cada indivíduo
    Selecionar indivíduos proporcionalmente ao fitness
    Aplicar cruzamento para gerar nova população
    Aplicar mutação com taxa definida
    Manter o melhor indivíduo (elitismo)
    Retornar o melhor indivíduo encontrado
```

## **4. Correspondência entre Pseudocódigo e Implementação**

Inicialização: função `create_individual()`. Avaliação: função `fitness()`. Seleção: `random.choices()` com pesos proporcionais ao fitness. Cruzamento: função `reproduce()`. Mutação: função `mutate()`. Elitismo: manutenção explícita do melhor indivíduo a cada geração. Interface e execução: implementadas com Streamlit.

## **5. Conclusão**

O algoritmo genético demonstrou ser adequado para o problema de otimização combinatória envolvendo múltiplos atributos e variáveis categóricas. A abordagem permite explorar diversas combinações possíveis e convergir para personagens com alto desempenho ofensivo.