



Algoritmos e Estruturas de Dados

Trabalho Final

A cidade de Nova York está um caos! O Destruidor e Krang uniram forças, e o Clã do Pé, junto com mutantes como Bebop e Rocksteady, estão executando vários planos malignos ao mesmo tempo. As Tartarugas Ninja estão sobrecarregadas.

Para organizar a defesa da cidade, Donatello decidiu criar um sistema de computador avançado, a "Central de Comando", para gerenciar as missões, catalogar os vilões e otimizar as operações do time. Sua tarefa é construir o software para esta Central de Comando e operacionalizar as batalhas que vem pela frente.

Sua missão é ajudar Donatello a desenvolver a Central de Comando, mas o desafio é grande. No esgoto, as máquinas são improvisadas e o poder de processamento é escasso. Por isso, o sistema precisa ser construído em C. Donatello sabe que essa é a única linguagem que lhe dará o controle manual sobre a memória e a performance bruta para criar um software ágil e robusto, capaz de rodar sem falhas em seus equipamentos limitados. O sistema deverá ser operado por uma interface de linha de comando (CLI), como se você estivesse no terminal do covil.

Descrição Funcional do Sistema:

1. Entidades do Sistema:



Tartaruga:

- o nome: Leonardo, Donatello, Raphael, Michelangelo
- o status: Disponível, Em Missão, Ferido.
- o nível: 1-10



Vilão: nome (Destruidor, Krang, Bebop, etc.), nível de habilidade (1 a 10), ponto fraco (texto)





- Há um [banco de dados](#) que as tartarugas vêm alimentando com os vilões mais frequentes nas últimas semanas. Você pode usar como referência, pois eles podem seguir tentando atacar, mas lembre-se que novos vilões podem surgir.
- 🍕 Missão: descrição (Ex: "Assalto ao Banco Central"), local (Ex: "Centro da cidade"), e um nível de ameaça (de 1 a 10).
 - Descrição de missões e seus respectivos níveis de ameaça são aleatórios, não há um padrão neles.
- 🍕 Equipamento: nome (Ex: "Katanas", "Bō", "Nunchakus", "Sai", "Bumerangue-Pizza", "Comunicador-Casco").

2. Módulos da Central de Comando:

Aqui estão as funcionalidades que sua Central de Comando deve ter. Pense cuidadosamente em como organizar os dados para que cada operação seja o mais eficiente possível. Em meio a tantos ataques, Donatello não deixou instruções sobre quais estruturas de dados devem ser utilizadas para gerenciar cada módulo da central.

🍕 Módulo 1: O Painel de Alertas de Missões

Novas missões são reportadas para a Central a todo momento. Elas chegam de fontes como April O'Neil ou de scanners da polícia.

O sistema precisa organizar essas missões pendentes de uma forma que as Tartarugas consigam sempre visualizar a descrição e o nível de ameaça primeiro. Se duas missões tiverem o mesmo nível de ameaça, a que chegou primeiro tem preferência.

Ao iniciar uma missão esta deve ser removida do painel de alertas.

🍕 Módulo 2: Os Arquivos de Vilões do Donatello

Donatello mantém um banco de dados sobre cada vilão que já enfrentaram.

O sistema precisa armazenar os perfis dos vilões (nome, nível de habilidade, ponto fraco).





Deve ser possível buscar informações sobre um vilão, pois as Tartarugas podem precisar consultar informações sobre um inimigo no meio de uma batalha.

Como novos vilões estão se unindo ao Clã do Pé, a central também deve permitir adicionar e remover vilões do arquivo.

Módulo 3: O Diário de Bordo do Mestre Splinter

Mestre Splinter exige um relatório de todas as missões concluídas para analisar o desempenho de seus filhos.

Toda vez que uma missão é finalizada, ela deve ser adicionada a um "Diário de Bordo". Mestre Splinter sempre pergunta coisas como: "Qual foi sua última vitória?", "Alguém se feriu?".

Módulo 4: O Arsenal e o Plano de Ação

Para cada missão iniciada, as Tartarugas precisam de um plano.

O sistema deve permitir que o jogador designe uma ou mais tartarugas disponíveis para a missão.

Além disso, o jogador deve poder montar uma lista de equipamentos do arsenal para levar àquela missão específica. A ordem dos equipamentos na lista não importa, mas deve ser fácil adicionar e remover itens.

Sobre as batalhas

A batalha não é algo que o jogador "joga" (pelo menos não precisa ser - é opcional), mas sim um evento que o programa resolve com base em uma estratégia definida na implementação. O processo é dividido em duas fases: Preparação e Resolução.

1. Fase de Preparação





Antes de a batalha ser resolvida, o jogador usa a Central de Comando para se preparar:

- 🍕 Escolhe a Missão
- 🍕 Consulta os Arquivos dos Vilões
- 🍕 Monta a Equipe
- 🍕 Seleciona o Equipamento (até dois)

2. Fase de Resolução

Uma vez que o jogador envia a equipe, o programa calcula o resultado instantaneamente.

- 🍕 "Força da Equipe": Calculada a partir da média da habilidade dos membros da equipe. Quando um membro da equipe está equipado com a sua arma favorita (Leonardo - Katanas; Donatello - Bō; Michelangelo - Nunchakus; Raphael - Sai) é somado uma habilidade bônus de +0.5 a força da equipe.

- 🍕 Vencedor (com um fator de sorte): O programa calcula o resultado com uma fórmula simples, adicionando um pequeno elemento aleatório para simular o caos de uma luta.

- Resultado Final = Força da Equipe - Nível de Desafio + Sorte (um número aleatório de -2 a +2) - Caos Mutante (número aleatório entre 1 e 0)

- 🍕 As Consequências (O resultado da batalha): Com base no Resultado Final, o sistema determina o que aconteceu e atualiza o estado do jogo.

- Se Resultado Final for alto (ex: > 3): Vitória Perfeita!
 - A missão é considerada um sucesso e movida para o Diário de Bordo do Mestre Splinter (a pilha de missões concluídas).
 - Todas as Tartarugas da equipe retornam com o status "Disponível" e aumentam seu nível em +0.3.
- Se Resultado Final for médio (ex: de 0 a 3): Vitória com Custo.
 - A missão é um sucesso e vai para o Diário de Bordo.





- Porém, uma das Tartarugas se feriu! Seu status muda para "Ferido" e ela fica indisponível pelas próximas 2 missões.
- Tartarugas que não se feriram aumentam seu nível em +0.2.
- Se Resultado Final for baixo (ex: < 0): Derrota!
 - A missão falhou. Ela não vai para o Diário de Bordo. (Pode até voltar para o painel de alertas).
 - Toda a equipe enviada retorna com o status "Ferido" e fica indisponível por um tempo (definido pelo programador). Não há aumento de nível nesse caso.

Critérios de avaliação:



- Código (80%)
 - O código desenvolvido pelos alunos será avaliado em função de sua corretude, completude, e organização.
 - Entregáveis:
 - README
 - Makefile (trabalhos que não façam uso do makefile terão a nota final reduzida em 50%).
 - arquivos com extensão .c e .h utilizados na implementação;
 - Pelo menos quatro das seguintes estruturas de dados devem ser implementadas e utilizadas no trabalho:
 - Lista (simplesmente/duplamente) encadeada
 - Filas
 - Pilhas
 - Heaps
 - Árvore binária (não pode ser completamente degenerada).
- Apresentação (20%):
 - O trabalho deverá ser apresentado em aula para a professora, sendo que a nota de apresentação (20%) de cada aluno é individual.
- A nota final do trabalho será calculada como a média harmônica entre os componentes de avaliação: Código (80%) e Apresentação (20%).





$$M_h = \frac{w_1 + w_2}{\frac{w_1}{n_1} + \frac{w_2}{n_2}}$$

Onde:

- w1 e w2 partes do trabalho (código = 80 e apresentação = 20)
- n1 é a nota obtida no Código.
- n2 é a nota obtida na Apresentação.

O trabalho poderá ser realizado em **duplas**. O formato de apresentação do trabalho deverá ser definido pelo docente responsável no dia definido para a mesma, não cabendo ao aluno escolher quem da dupla (quando houver) é responsável por cada parte da apresentação. Em caso da realização do trabalho em dupla, ambos os alunos são responsáveis pelo código implementado e pelo domínio do conhecimento acerca do mesmo. No momento da avaliação do trabalho, pode ser solicitado aos alunos realizar alterações no código para fins de validação do conhecimento. Observação: caso o aluno não demonstre conhecimento sobre o código entregue para avaliação, a nota geral do trabalho tenderá a zero.

→ Plágio: Trabalhos com similaridade igual ou superior a 70% a outros (inclusive de outros semestres) receberão nota 0 ;)

Entrega: até 12 de agosto de 2025 às 23h59.

Apresentações: 13 e 14 de agosto de 2025 às 23h59.

➤ Submissões realizadas fora do prazo estipulado **não** serão consideradas para avaliação.u

