

# SME0827 - Estruturas de Dados

## Primeiro semestre de 2021

**Professora:** Marina Andretta (andretta@icmc.usp.br)  
**Monitor:** Tales S. Somensi (somenzi@usp.br)

### Trabalho - Jogo de Ficção Interativa

## 1 Grupos

- O trabalho será feito em grupos de 4 pessoas.
- No e-disciplinas há um tópico para que os grupos sejam escolhidos. Caso alguém não escolha seu grupo, a professora irá fazê-lo de forma aleatória (veja mais detalhes no e-disciplinas).

## 2 Enunciado

Os jogos de Ficção Interativa (Interactive Fiction ou IF) são ambientes de simulação totalmente baseados em texto, em que um jogador emite comandos de texto para efetuar mudanças no ambiente e progredir na história. Segundo estudos, os humanos utilizam competências como compreensão da linguagem, raciocínio de senso comum e dedução para inferir o contexto e os objetivos de jogos.

O objetivo deste trabalho é implementar um jogo de Ficção Interativa, em Python 3, baseado em texto. A execução do jogo deve ocorrer na linha de comandos (terminal).

O jogo a ser implementado se trata de um enigma em que o personagem principal começa preso em um determinado ambiente. O objetivo é conseguir chegar em um ambiente “final”. O personagem pode mover-se, observar o ambiente e cada objeto presente nele. Alguns objetos são interativos, eles podem ser coletados para o inventário do personagem e podem ser utilizados através de uma ação. O jogador precisa encontrar meios de utilizar os objetos para abrir portas e compartimentos, e assim progredir no seu objetivo. A mecânica é muito similar ao jogo “Zork”, disponível em <https://cutt.ly/DnfHh7y>. A Figura 1 mostra um exemplo.

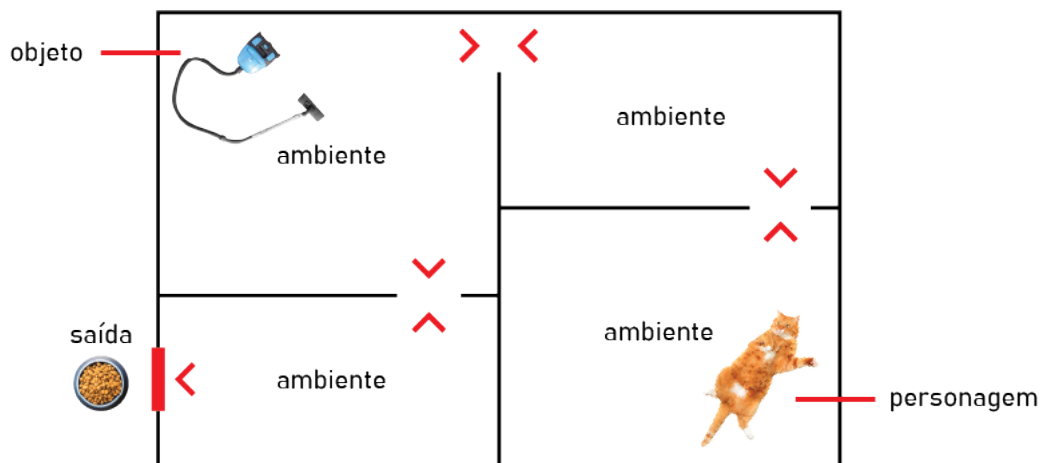


Figure 1: Exemplo

A história e narrativa são livres, cada grupo pode criar a história que quiser (desde que não haja ofensas). O jogo pode ser em português ou em inglês. Alguns termos podem ser encurtados

(por exemplo, pode-se usar “move south” ao invés de “move to south”), desde que isso fique claro para o usuário.

O jogo deve satisfazer algumas condições:

1. deve haver quatro ou mais **ambientes**;
2. deve haver um **inventário** para coleta de objetos pelo jogador;
3. cada ambiente deve conter três ou mais **objetos interativos** (que se pode colocar no inventário) e três ou mais objetos não interativos;
4. cada objeto interativo deve conter ao menos uma **ação** correspondente;
5. todos os ambientes e objetos devem conter uma descrição que poderá ser acessada através de uma ação (por exemplo, ação “ver” ou “look at”);
6. pelo menos oito objetos interativos devem ser utilizados para vencer o jogo;
7. no início, o jogador deve ser introduzido à narrativa (por exemplo: “Você é o brinquedo favorito do Andy e está perdido no sótão ...”);
8. ao digitar **help** ou **h** no terminal, deve ser apresentada uma breve descrição das ações disponíveis e/ou instruções para o jogador.

### 3 Instruções

O trabalho será composto de uma implementação (em Python 3), um relatório com uma breve descrição do que foi feito (em PDF) e uma apresentação. O código em Python 3 (.py) e o relatório (.pdf) devem ser enviados pelo e-disciplinas em um único arquivo .zip. Veja mais detalhes no e-disciplinas.

Qualquer integrante do grupo pode fazer o envio do trabalho. Apenas o último arquivo enviado será considerado.

A seguir, veja as instruções do que deve estar em cada uma das partes a serem feitas:

#### 1. Código

Deve ser implementado em Python 3. O código deve ser organizado, claro e legível. Deve incluir comentários objetivos sobre a lógica utilizada. Os nomes e números USP de todos os integrantes deverão constar no início do código.

#### 2. Relatório

Deve conter uma breve descrição da história, breves instruções de jogo, explicações sobre a lógica utilizada na implementação do código e sobre quais conceitos da disciplina foram utilizados na implementação (e porque foram feitas essas escolhas). Podem ser utilizados diagramas para explicar as estruturas de dados usadas. O relatório deve ter, no máximo, 4 páginas.

Os nomes e números USP de todos os integrantes deverão constar no início do relatório.

#### 3. Apresentação

Deve apresentar brevemente a história/narrativa escolhida, quais ambientes, objetos e ações fazem parte do jogo. Explicar quais conceitos de estrutura de dados foram utilizados e sua finalidade dentro do código implementado. Podem ser incluídas outras informações que o grupo desejar. A apresentação deve ter até 8 minutos.

Tanto a entrega do código e relatório, como a apresentação do trabalho, serão no dia 22/07. Mais detalhes podem ser vistos no e-disciplinas.

## 4 Avaliação

O trabalho vale 10 pontos. A avaliação levará em conta 3 partes:

- Código (C): **7** pontos, nota do grupo. Serão avaliadas:
  - implementação dos conceitos apresentados na disciplina;
  - legibilidade do código implementado;
  - consistência do jogo implementado (bugs, falhas).
- Relatório (R): **3** pontos, nota do grupo. Serão avaliados:
  - conteúdo (ver Seção 3).
  - riqueza de informações;
  - clareza.
- Apresentação (A): **1** ponto, nota individual. Serão avaliadas:
  - participação individual na apresentação;
  - resposta à(s) pergunta(s);
  - clareza.

A nota do trabalho será calculada da seguinte forma:

$$Nota = \begin{cases} (C + R)A, & \text{se código e relatório forem entregues} \\ 0, & \text{caso contrário} \end{cases}$$

**ATENÇÃO:** Trabalhos iguais (ou muito parecidos) de grupos diferentes receberão nota 0.

## 5 Sugestões

Para facilitar o entendimento do trabalho proposto, recomenda-se:

- Jogar o game “Rime - room escape game -”, disponível gratuitamente no Google Play e na App Store. 30 minutos são suficientes para entender a mecânica do jogo;
- Fazer o tutorial disponível em <https://cutt.ly/6ndpEiR>. É rápido de ser concluído e dá uma boa noção sobre como implementar um jogo desse tipo;
- Ao final, o resultado será parecido com o jogo “Zork”. Disponível em <https://cutt.ly/DnfHh7y>.

## 6 Observação

Quaisquer dúvidas sobre o trabalho podem ser respondidas pelo Tales (somenzi@usp.br) ou diretamente com a professora.