



PROJETO DA DISCIPLINA

---

# Raciocínio Computacional

*Disciplina EAD*

---



# Sumário

<b>SEMANA 1</b> .....	<b>2</b>
<b>Projeto da Disciplina</b> .....	<b>2</b>
<b>Atividade Formativa</b> .....	<b>3</b>
Fórum: Pseudocódigo, por onde começar? .....	3
<b>SEMANA 2</b> .....	<b>3</b>
<b>Atividade Formativa</b> .....	<b>3</b>
<b>SEMANA 3</b> .....	<b>5</b>
<b>Atividade Formativa</b> .....	<b>5</b>
<b>SEMANA 4</b> .....	<b>6</b>
<b>Atividade Somativa 1</b> .....	<b>6</b>
Rubrica de correção de atividade.....	6
<b>Atividade Formativa</b> .....	<b>7</b>
<b>SEMANA 5</b> .....	<b>8</b>
<b>Atividade Formativa</b> .....	<b>8</b>
<b>SEMANA 6</b> .....	<b>9</b>
<b>Atividade Formativa</b> .....	<b>9</b>
<b>SEMANA 7</b> .....	<b>10</b>
<b>Atividade Formativa</b> .....	<b>10</b>
<b>SEMANA 8</b> .....	<b>11</b>
<b>Atividade Somativa 2</b> .....	<b>11</b>
Rubrica de correção de atividade.....	12



# Projeto da disciplina

Raciocínio Computacional – EAD

## SEMANA 1

---

### PROJETO DA DISCIPLINA

Entre a parceria da Galápagos jogos e uma empresa de desenvolvimento de jogos digitais você foi contratado para realização de um projeto de implementação de um jogo. O jogo consiste na criação de uma versão do jogo de tabuleiro Zombie Dice para o computador. Neste jogo o jogador é zumbi faminto por cérebros, que deve comer mais cérebros que seus amigos, porém, cuidando com os tiros de espingarda. Para a criação deste jogo você precisará integrar todo o seu conhecimento em dados primitivos, dados homogêneos e heterogêneos (tuplas, listas e dicionários), operadores lógicos relacionais e aritméticos, estruturas condicionais (if, else e elif), estruturas de repetição (for e while), funções, métodos e classe.

**Dinâmica do Jogo:** O jogo é composto por um tubo que armazena 13 dados de 6 faces. Existem 3 tipos de dados diferentes: verdes, vermelhos e amarelos. Os dados possuem 3 símbolos diferentes (tiro, cérebro e passos). Os dados vermelhos são os mais difíceis, possuem um número maior de faces com o símbolo tiro. Os verdes são os mais fáceis, o maior número de faces do dado é de cérebro. Os dados amarelos são intermediários as faces dos dados possuem a mesma quantidade cérebros, tiros e passos. Em um rodada, cada jogador tem um turno para jogar. No turno, o jogador deve pegar aleatoriamente 3 dados do tubo e lançar, sempre os três dados juntos. Quando a face do dado cair virada com símbolo para "cérebro", é porque o jogador comeu um cérebro. Caso a face do dado cair virada com o símbolo para "passos", a vítima fugiu, se o jogador decidir continuar a lançar os dados em seu turno, os dados que caíram com a face para "passos" deverão ser lançados novamente, para lançar novamente os três dados o jogador completa a quantidade retirando outros dados do tubo, sempre lançando os três dados. O objetivo é comer 13 cérebros para vencer o jogo, no entanto existem os tiros de espingarda, se por acaso o jogador levar três tiros de espingarda ele perde e sai do jogo. No seu turno após lançar os dados o jogador pode decidir parar ou continuar a jogar. Caso decida parar você contabiliza os cérebros para próxima jogada, já os dados de tiro são zerados para próxima rodada.

**Regras do Jogo:** Para facilitar o desenvolvimento do jogo as regras estão disponibilizadas no arquivo disponível no ambiente virtual de aprendizagem.



## ATIVIDADE FORMATIVA

Vamos iniciar o desenvolvimento do jogo? Nesta semana você deverá estruturar em pseudocódigo a dinâmica do jogo, identificando os passos do jogo e colocando-os em sequência, ou seja, estruturar o algoritmo em pseudocódigo. Primeiramente faça um esboço do pseudocódigo no papel ou se preferir no bloco de notas.

Quando possível utilize as seguintes palavras: “INÍCIO”, “FIM”, “SE”, “ENTÃO”, “SENÃO”, “ENQUANTO”, “PARA”, “DE”, “ATE”, “FAÇA”.

Diferencie os tipo de variáveis (str, int e float) e identifique onde as variáveis podem ser substituídas no pseudocódigo. O pseudocódigo será solicitado na entrega da primeira fase do projeto (prevista para quarta semana). Deste modo, a versão final do seu pseudocódigo deve estar no formato de arquivo .txt ou .doc, adicione o seu nome e o curso no início do arquivo.

Agora traduza seu pseudocódigo para linguagem de programação Python. Esta atividade pode ser desenvolvida no **repl.it** (<https://repl.it/>) ou **PyCharm** (<https://www.jetbrains.com/help/idea/product-educational-tools.html>). Embora o **repl.it** seja uma plataforma que roda online, sem a necessidade de instalação local no computador, o **PyCharm** é uma ferramenta completa, contendo diversos recursos que podem auxiliar ao longo do projeto.

Você precisará fazer a interação com entrada e saída de dados utilizando os comandos *print* e *input* e deverá manipular mensagem de entrada e saída para os jogadores.

Finalmente precisará formatar as mensagens do jogo utilizando o conceito de placeholder e da função *format*.

Para a realização dessas tarefas estude a **Unidade 1: Introdução à programação Python**, principalmente, assista ao vídeo **Formatação da saída de dados (placeholder e format)**.

### Fórum: Pseudocódigo, por onde começar?

Você tem dúvidas de como começar a elaborar seu pseudocódigo? Compartilhe, postando no fórum. Já tem experiência com pseudocódigo? Ajude os colegas, contribua com os seus conhecimentos!

## SEMANA 2

---

## ATIVIDADE FORMATIVA

Nesta semana você iniciará com os comandos condicionais, nesta etapa você deverá estabelecer algumas condições que podem ser verificadas na lógica do jogo, aplicando os comandos *if*, *elif* e *else*.

Para isso, estude a **Unidade 2: Estrutura condicionais**, principalmente, assista ao vídeo **Gerando números randômicos em Python**, que apresenta dicas importantes que irão te ajudar na utilização da biblioteca



random para trabalhar com números aleatórios. Com isso, você será capaz de gerar sequências aleatórias para a criação dos dados.

Assim, você deverá criar os dados neste momento utilizando um objeto do tipo *string*. Esta *string* deve conter seis caracteres para simular cada face do dado. Para identificar as faces dos dados segue abaixo especificação:

- 6 Dados verdes: "CPCTPC"
- 4 Dados amarelos: "TPCTPC"
- 3 Dados vermelhos: "TPTCPT"

Onde, o caractere "C" na string corresponde ao cérebro, caractere "P" são os passos e por fim o "T" é o tiro.

Neste momento você deve utilizar a string de forma aleatória, poderá utilizar a função `random.choice()` ou se preferir poderá trabalhar com o índice da string.

Perceba que você começará a aplicar conceitos de operadores lógicos e relacionais, sendo capaz de efetuar comparações entre os caracteres que foram sorteados aleatoriamente.

Assim, utilizando os dados, você já consegue começar a trabalhar com a lógica do jogo. Consegue definir blocos de código utilizando as estruturas condicionais, ou seja, quando o jogador cair em cada um das faces do dado, poderá incrementar um contador (cérebro, passos e tiro ).

Checklist do projeto	Status
Utilizou as estruturas condicionais (if, else e elif)	
Utilizou os operadores lógicos e aritméticos	
Criou os dados utilizando uma string (6 verdes, 4 amarelos e 3 vermelhos)	
Adicionou a entrada para receber o jogadores	
Implementou a rotina para sortear os dados (selecionar três dados)	
Implementou a rotina para jogar os dados (sortear a face de um dos dados)	
Implementou a entrada para receber a quantidade de jogadores	



Efetua checagem se foi adicionado pelo menos dois jogadores	
Criou as variáveis para contabilizar os tiros, cérebros e passos	
Incrementa as variáveis quando um dos dados é lançado	

## SEMANA 3

### ATIVIDADE FORMATIVA

Nesta semana você deverá moldar o protótipo inicial do jogo, ou seja, elaborar um esqueleto do jogo, considerando:

- Interação com o jogador, entradas e saídas de dados utilizando as estruturas de repetição.
- Criação de uma opção para lançar os três dados aleatoriamente, verificar qual foi o resultado de cada um dos dados.
- Criação das variáveis para contabilizar os tiros, cérebros e passos.

Para realizar essas tarefas, estude a **Unidade 3: Estruturas de repetição**.

Checklist do projeto	Status
Utilizou as estruturas repetição (for e while)	
Realiza a Interação do jogador utilizando a estrutura de repetição	
Implementou a rotina para sortear os dados dentro da estrutura de repetição	
Implementou a rotina para jogar os dados dentro da estrutura de repetição	
Criou as variáveis para contabilizar os tiros, cérebros e passos adicionando na estrutura de repetição	
Incrementa as variáveis quando um dos dados são lançados	



Verifica o resultado dos três dados lançados	
--	--

## SEMANA 4

### ATIVIDADE SOMATIVA 1

Nesta semana você fará a entrega do protótipo inicial do jogo que foi realizado durante as semanas anteriores. Este protótipo deve apresentar:

1. Construção de algoritmos em pseudocódigo
2. Manipulação de variáveis e constante
3. Uso de estruturas condicionais (if, elif e else) na implementação em Python
4. Uso de operadores lógicos e relacionais
5. Uso de estruturas de repetição (for e while)

#### Especificação de entrega:

Você deve entregar dois arquivos o pseudocódigo referente a atividade da primeira semana e a implementação inicial do protótipo.

O pseudocódigo deve ser entregue em um arquivo no formato .txt ou .doc, este corresponde ao esquema proposto na primeira semana.

O código deve estar devidamente comentado, estruturado em um único arquivo na extensão padrão do Python (".py"). Este deve conter o nome do estudante e curso adicionados no cabeçalho do arquivo em um bloco de comentário. O arquivo deve estar compactado no formato ".zip" e postado no Blackboard até a data limite.

#### Rubrica de correção de atividade

CRITÉRIOS	Autônomo (100% a 90%)	Capaz (89% a 70%)	Aprendiz (69% a 40%)	Em desenvolvimento (39% a 0)
Construção de algoritmos em pseudocódigo (20%)	Constrói adequadamente algoritmos Python e pseudocódigos.	Constrói corretamente algoritmos em pseudocódigo, apresenta um pouco de dificuldades para implementar o algoritmo em Python.	Identifica e define a sequência de passos para construção de algoritmos.	Tem dificuldades em compreender e implementar algoritmos em pseudocódigo.



Manipulação de variáveis e constante (10%)	Emprega, adequadamente, as variáveis e constante com os tipos corretos de dados.	Manipula corretamente as variáveis e constante, ainda não aplica a convenção de boas práticas da linguagem.	Sabe identificar quando utilizar os diferentes tipos de dados, mas tem dificuldade de efetuar conversão entre os dados.	Tem dificuldades para diferenciar os tipos de dados.
Uso de estruturas condicionais (if, elif e else) na implementação em Python (25%)	Emprega, adequadamente, as estruturas condicionais com os comandos if, elif e else.	Manipula corretamente as estruturas condicionais, mas ainda não está familiarizado com as condições aninhadas.	Sabe identificar as estruturas condicionais (if, elif e else), ainda apresenta dificuldades em estruturar a lógica nos condicionais.	Tem dificuldades em utilizar as estruturas condicionais.
Uso de operadores lógicos e relacionais (20%)	Emprega, adequadamente, os operadores lógicos e relacionais.	Manipula corretamente os operadores relacionais, ainda tem dúvidas na manipulação de condicionais compostos com os operadores and e or.	Sabe identificar os operadores lógicos e relacionais.	Tem dificuldades em utilizar os operadores lógicos e relacionais.
Uso de estruturas de repetição (for e while) na linguagem Python (25%)	Emprega, adequadamente, as estruturas de repetição com o comandos while e for.	Manipula corretamente as estruturas condicionais, mas ainda tem dificuldades em estabelecer critérios de parada para o comando while.	Consegue diferenciar o uso das estrutura de repetição (while e for).	Tem dificuldades em utilizar as estruturas de repetição.

## ATIVIDADE FORMATIVA

Após a entrega do protótipo inicial, você dará sequência no desenvolvimento dele, ou seja, irá implementar a lógica do jogo, utilizando a estrutura de lista.

Assim, você deverá implementar:

- Uma lista para armazenar os jogadores. Observe que para iniciar o jogo é necessário no mínimo dois jogadores.





- Uma lista para representar o tubo, isto é, copo onde serão colocados os treze dados inicialmente, lembre-se de que os dados devem ser retirados aleatoriamente do copo (remover da lista), posteriormente lançados aleatoriamente.
- Por fim, uma lista para armazenar a pontuação dos jogadores (cérebros, passos e tiros), caso o jogador decida encerrar seu turno, antes de levar três tiros ou comer 13 cérebros. Ao encerrar o turno, os tiros deverão ser zerados para próximo turno de cada jogador.

Para realizar essas tarefas, estude a **Unidade 4: Listas**.

## SEMANA 5

### ATIVIDADE FORMATIVA

Nesta semana você deverá modificar a estrutura dos dados. Inicialmente eles foram criados com uma estrutura do tipo string, a ideia é utilizar as tuplas no lugar da string.

Importante! A partir desta semana você já deve ter implementado a lógica e a jogabilidade do jogo.

Para realizar essas tarefas, estude a **Unidade 5: Estruturas de dados: tuplas e dicionários**.

Checklist do projeto	Status
Modificou a estrutura dos dados de string para tupla	
Utilizou uma estrutura para armazenar os jogadores	
Utilizou uma estrutura para armazenar os dados do jogo (simular o copo)	
Remove os dados da lista em cada jogada	
Armazena a pontuação dos jogadores na lista (quantidade de cérebros e tiros)	
Reinicia a lista que armazena os dados sempre que o turno de um jogador termina (adicionado novamente os 13 dados no copo)	
Reinicia a pontuação de tiros na lista sempre que o jogador finalizar seu turno	



Imprime na saída do console quais os dados que estão armazenados no copo em cada uma das jogadas	
Imprime na saída do console a cor de cada um dos três dados sorteados	
Imprime na saída do console a face de cada um dos dados sorteados	
Implementou a lógica e a jogabilidade do jogo	

## SEMANA 6

---

### ATIVIDADE FORMATIVA

Nesta semana você deverá modificar o protótipo do jogo, ou seja, deverá modularizar o código por meio de funções.

Você deverá criar funções para:

- lançar os dados;
- mostrar os dados;
- mostrar os dados no copo;
- pegar cada um dos tipos de dados (verde, amarelo, vermelho);
- inicializar os dados a cada turno; e
- verificar score (cérebros, tiros e passos).

Fique atento! No final desta semana, você já deve estar com o jogo funcionando completamente para a segunda entrega.

Para realizar essas tarefas, estude a **Unidade 6: Funções**.

Checklist do projeto	Status
Esta utilizando funções para modularizar o código	
Criou uma função para pegar cada um dos dados (verde, amarelo e vermelho)	



Criou uma função para inicializar os dados no copo	
Criou uma função para remove os dados do copo (remover da lista em cada jogada)	
Criou uma função para lançar os dados (sortear a face do dado )	
Criou uma função para mostrar os dados que estão no copo	
Criou uma função para mostrar a cor dos que foram sorteados	
Criou uma função para apresentar a pontuação do jogo	

## SEMANA 7

### ATIVIDADE FORMATIVA

Nesta semana você otimizará o seu código, que já deverá estar modularizado com as funções.

Para isso, você aplicará as boas práticas de programação para otimizar as funções com foco em reutilização de código.

Para realizar essas tarefas, estude a **Unidade 7: Funções: aplicação em problema de contexto real.**

Checklist do projeto	Status
Efetuiu otimização no código seguindo boas práticas	
Criou uma função para verificar se o jogador jogador ganhou ou perdeu	
Criou uma função para verificar se o jogador deseja parar ou continuar a jogar no seu turno.	
Otimizou as funções criadas anteriormente	



Está reutilizando o código por meio das funções	
Otimizou o fluxo de execução principal do programa	

## SEMANA 8

---

### ATIVIDADE SOMATIVA 2

Nesta semana você conhecerá os conceitos básicos de orientação objeto (apenas uma noção dos conceitos, pois serão aprofundados em outra disciplina).

Neste momento o protótipo deverá ser estruturado em uma classe. Você trará as funções implementadas anteriormente para dentro da classe, efetuando as modificações necessárias.

Até o final desta semana você deverá finalizar e entregar o protótipo, considerando:

- Manipulação da estrutura lista
- Manipulação da estrutura tuplas
- Uso de funções para modularizar o código
- Uso de funções visando o reuso e boas práticas
- Uso de classes e métodos

Você deverá entregar a versão final do protótipo e um relatório compactados em um único arquivo no formato “zip”, postado no Blackboard até a data limite.

Na versão final do protótipo o jogo deve estar concluído, contemplando todas as funcionalidades especificadas no projeto. O protótipo deve ser implementado na linguagem de programação Python. Todo código deve estar devidamente comentado, sendo estruturado em um único arquivo na extensão padrão do Python (“.py”). Adicionar no cabeçalho do arquivo Python o seu nome e curso (utilizar um bloco de comentário).

O relatório deve ser entregue no formato “.pdf”, deve conter uma descrição do protótipo do jogo, ou seja, descrever as funcionalidades do jogo que foi implementado – deve demonstrar a interação de dois jogadores. Para cada interação do jogo você deve coletar um *printscreen* da saída do console, adicionar no relatório e descrever o que foi realizado.

Para realizar essas tarefas, estude a **Unidade 8: Métodos e Classes em Python**.



## Rubrica de correção de atividade

CRITÉRIOS	Autônomo (100% a 90%)	Capaz (89% a 70%)	Aprendiz (69% a 40%)	Em desenvolvimento (39% a 0)
Manipulação da estrutura lista <b>(25%)</b>	Emprega, adequadamente, o uso de listas, e conhece as funções para manipulação de lista.	Manipula corretamente as listas, consegue inserir e remover elementos da lista, ainda não conhece todas as funções da estrutura lista.	Sabe identificar a estrutura de lista mas tem dificuldades para inserir e remover elementos da lista.	Tem dificuldades em saber quando e como utilizar a estrutura lista.
Manipulação da estrutura tuplas <b>(10%)</b>	Emprega, adequadamente, o uso da estrutura tupla, e sabe como iterar com os elementos da tupla.	Manipula corretamente tuplas, mas ainda fica na dúvida quando utilizar tuplas ou listas.	Sabe identificar as diferenças entre a estrutura lista e tupla.	Tem dificuldades em saber quando e como utilizar a estrutura tupla.
Uso de funções para modularizar o código <b>(25%)</b>	Emprega, adequadamente, as funções, trabalha com passagem de parâmetros e retorno nas funções.	Manipula funções corretamente, consegue efetuar passagem de parâmetros e retorno na função, ainda não aplica as boas práticas para reuso em funções.	Sabe identificar funções no código, consegue criar funções sem passagem de parâmetros.	Tem dificuldades em criar funções.
Uso de funções visando o reuso e boas práticas <b>(20%)</b>	Emprega, adequadamente, as funções focado no reuso e noções de boas práticas.	Manipula funções corretamente, mas ainda não aplica as boas práticas voltadas a reuso de código.	Sabe utilizar funções mas ainda tem dificuldades com passagem de parâmetros e retorno em funções.	Tem dificuldades em utilizar funções.
Uso de classes e métodos <b>(20%)</b>	Emprega, adequadamente, o uso de classes e métodos em Python.	Manipula corretamente classes e métodos, têm dificuldades em visualizar o que é objeto e instância.	Sabe identificar classes e métodos em Python, mas tem dificuldades de estruturar o código com uso de classes e funções.	Tem dificuldades de utilizar classes e métodos.



**PUCPR**  
GRUPO MARISTA