

INF100 – Introdução à Programação I
Roteiro da Aula Prática 07 – 29/09 a 01/10/2025
Comando repetitivo while - Valor: 2 pontos

Instruções

Baixe o arquivo **p07.py**. Nome do arquivo a ser entregue: **p07.py**

Definição do problema a ser resolvido:

Jogo de adivinhação

Uma brincadeira de criança que consiste em tentar adivinhar um número qualquer em um intervalo, definido. Nas brincadeiras, a criança que pergunta diz o intervalo (“Escolha um número de 1 a 5”) e ela própria diz se a resposta foi correta, após ouvir o palpite de outra criança. Como apenas a criança que faz a pergunta sabe qual é a resposta (o número secreto), dependendo da faixa etária, pode ficar a impressão de que está havendo trapaça.

Para que não haja trapaça, o sorteio do número secreto, a conferência e contagem dos palpites válidos serão feitos por um programa Python. Vamos explicar o funcionamento do programa com um exemplo: suponha que o número sorteado foi o **33**, com o intervalo inicial **[1, 100]**.

Se o primeiro palpite do usuário for **50**, o programa deverá contar como palpite válido (estava no intervalo) e informar que o palpite está errado, explicitando que o número sorteado é **MENOR** do que o palpite dado. Um novo intervalo será definido em função do palpite dado: novo intervalo **[1, 49]**. Sempre que pedir um novo palpite, o programa informa o intervalo atual.

Se o segundo palpite for **51**, o programa deverá informar que o valor é inválido (está fora do intervalo atual **[1, 49]**), não contar o palpite inválido e solicitar um novo palpite.

Caso o novo palpite seja **25**, o programa deverá contar como palpite válido (estava no intervalo **[1, 49]**) e informar que o palpite está errado, explicitando que o número sorteado é **MAIOR** do que o palpite dado. Um novo intervalo será definido em função do último palpite dado: novo intervalo **[26, 49]**.

Caso o próximo palpite seja **33**, o programa deverá terminar, informando que o número sorteado foi o 33 e que o usuário acertou na 3a. tentativa, pois APENAS os palpites que estavam dentro do intervalo válido, no momento do palpite, são contados.

Primeira Tarefa:

O programa p07.py já faz algumas poucas ações para implementar a solução para o problema apresentado. Dessa forma, a primeira tarefa é completar o programa Python (**p07.py**) para que ele faça todas as ações descritas no algoritmo apresentado na próxima página.

ATENÇÃO: as partes em negrito no algoritmo já estão implementadas no código

Algoritmo do Jogo de adivinhação.

- importar a biblioteca random (linha 9);
- definir um intervalo com limite inferior igual a 1 e limite superior igual a 100 (linhas 10-11);
- mensagem inicial do programa (linha 12);
- definir a semente de geração de números aleatório igual a 0 (zero) (linha 13);
- sorteio de um número aleatório no intervalo 1 a 100 (linha 14);
- criar uma variável para contar o número de palpites válidos dados pelo usuário do programa, com valor inicial igual a zero.
- Executar de maneira repetitiva os seguintes passos (while True da linha 22):
 - ler um palpite do usuário (linha 24).
 - validar o palpite lido, que deve estar no intervalo atual. Enquanto o palpite for inválido, o programa deverá dar uma mensagem de erro informando o intervalo atual, para orientar o usuário ([veja o terceiro exemplo](#))
 - contar o palpite válido
 - testar o palpite dado pelo usuário e executar uma de 3 ações possíveis:
 - 1) quando o palpite do usuário for igual ao número sorteado, quebrar (break) o comando repetitivo, ou;
 - 2) quando o palpite for maior do que o número sorteado, atualizar o limite superior para o valor do palpite menos 1.
 - 3) quando o palpite for menor do que o número sorteado, atualizar o limite inferior para o valor do palpite mais 1.
 - Após sair do while-True, imprimir o resultado. Esta parte está parcialmente pronta. **O primeiro print está pronto (linha 53).** Implemente o segundo print para informar o número de tentativas válidas.

Após ser completado corretamente, o seu programa deverá exibir saída como a mostrada a seguir:

1o. Exemplo:

```
>>> ===== RESTART: /home/goulart/Documentos/Disciplinas/INF100/2025-1/Prati
      Será sorteado um número de 1 a 100 e você deverá descobrir qual...!
      Qual é o seu palpite [1, 100]? 40
      Dica número 1: o número sorteado é maior que 40.
      Qual é o seu palpite [41, 100]? 60
      Dica número 2: o número sorteado é menor que 60.
      Qual é o seu palpite [41, 59]? 50
      Número sorteado: 50
      Você acertou na 3a tentativa
>>>
```

2o. Exemplo (2 execuções consecutivas):

```

>>> ===== RESTART: /home/goulart/Documentos/Disciplinas/INF100/2025-1/Pra1
    Será sorteado um número de 1 a 100 e você deverá descobrir qual...!
    Qual é o seu palpite [1, 100]? 10
    Dica número 1: o número sorteado é maior que 10.
    Qual é o seu palpite [11, 100]? 9
    Palpite fora do intervalo [11, 100]
    Qual é o seu palpite [11, 100]? 90
    Dica número 2: o número sorteado é menor que 90.
    Qual é o seu palpite [11, 89]? 91
    Palpite fora do intervalo [11, 89]
    Qual é o seu palpite [11, 89]? 50

    Número sorteado: 50
    Você acertou na 3a tentativa
>>> ===== RESTART: /home/goulart/Documentos/Disciplinas/INF100/2025-1/Pra1
    Será sorteado um número de 1 a 100 e você deverá descobrir qual...!
    Qual é o seu palpite [1, 100]? 50

    Número sorteado: 50
    Você acertou na 1a tentativa
>>>

```

Observe que o número sorteado foi sempre 50... e, neste caso, o jogo ficaria meio sem graça.

Segunda tarefa:

Inclua mais um comando repetitivo while para que o programa pergunte ao usuário se ele deseja jogar mais uma vez, após ele acertar o número sorteado. Quando a resposta for sim, um novo número deverá ser sorteado, o intervalo deverá voltar a ser [1, 100] e o contador do número de palpites deverá voltar a ser 0 (zero). A seguir um exemplo de execução após esta modificação:

```

>>> = RESTART: /home/goulart/Documentos/Disciplinas/INF100/2025-1/Praticas/Port
/p08/p08-solucao2.py

    Será sorteado um número de 1 a 100 e você deverá descobrir qual...!
    Qual é o seu palpite [1, 100]? 0
    Palpite fora do intervalo [1, 100]
    Qual é o seu palpite [1, 100]? 50

    Número sorteado: 50
    Você acertou na 1a tentativa
    Deseja jogar outra vez? (S/N): s

    Novo número entre 1 a 100 foi sorteado e você deverá descobrir qual...!
    Qual é o seu palpite [1, 100]? 101
    Palpite fora do intervalo [1, 100]
    Qual é o seu palpite [1, 100]? 98

    Número sorteado: 98
    Você acertou na 1a tentativa
    Deseja jogar outra vez? (S/N): s

    Novo número entre 1 a 100 foi sorteado e você deverá descobrir qual...!
    Qual é o seu palpite [1, 100]? 54

    Número sorteado: 54
    Você acertou na 1a tentativa
    Deseja jogar outra vez? (S/N): n

    Obrigado por experimentar o nosso jogo!
>>>

```

Comentários finais:

Observe que o usuário acertou sempre na primeira tentativa - embora tenha um palpite fora do intervalo, talvez para despistar. Cuidado, o dono de um programa de sorteio pode trapacear...

A trapaça consistiu em executar o sorteio repetidas vezes (com a semente igual a 0) e anotar os números sorteados: 50, 98, 54, 6, 34, 66, 63, 52, 39...

Por isso, você NUNCA deve fazer apostas contra um programa de computador, pois você vai perder!!! Do outro lado está a pessoa/empresa que programou o jogo para que, em algum momento, você perca. É só questão de tempo.

Com uma semente fixa, os números gerados são chamados de números pseudo-aleatórios. E usar números pseudo-aleatórios é muito bom quando você quer que o comportamento do programa seja determinístico, ao invés de ser aleatório. Como por exemplo para testar a mesma situação de sorteio dos exemplos deste roteiro.

Você deve deixar a semente igual a 0 no seu código, pois na correção, vamos usar os mesmos exemplos deste roteiro.

Deve ser entregue APENAS 1 programa p07.py com a sua versão final

 **A saída do seu programa deve obedecer à formatação exata mostrada nos exemplos.**

 **Não esqueça de preencher o cabeçalho com seus dados e uma breve descrição do programa.**

Após certificar-se que seu programa está correto, envie o arquivo do programa fonte (**p07.py**) através do sistema do LBI.

A entrega deverá ser feita até 4a. feira, 01/10/2025, às 23h59 no endereço:

<http://linux-server.lbi.ufv.br>