

EP2 - JOGO DOS BITS

Exercício-Programa 2 (EP2)

Data de entrega: **até 8/outubro/22 - 23h59 (sábado)**

Este segundo EP estende alguns conceitos utilizados no primeiro EP. Além das ferramentas como o módulo *random* e uso de *strings*, ele demanda conhecimentos sobre representações binárias de números inteiros (mudança de base). Neste EP você deve resolver duas tarefas sobre o Jogo dos Bits. Você deve usar um gerador de bits aleatórios, como foi feito no EP1 (para simular jogadas de uma moeda).

Tarefa 1

Alice, Beto e Casemiro são três amigos que gostam muito de jogos. Recentemente, eles descobriram o *Jogo dos Bits* (que pode ser jogado com lápis, papel e uma moeda). O jogo é o seguinte:

Dois jogadores, Alice e Beto, escolhem cada um uma sequência binária de comprimento k (supor $3 \leq k \leq 8$). Casemiro começa então a gerar bits aleatórios. A cada bit sorteado, Casemiro concatena o mesmo na sequência gerada até o momento. No instante em que a sequência de Alice ou de Beto aparecer como uma subsequência contínua (da sequência gerada por Casemiro), considera-se que a rodada terminou, e vence a rodada o/a jogador(a) cuja sequência apareceu primeiro.

Dada a natureza aleatória do jogo, é mais natural fazer várias rodadas e determinar a probabilidade de Alice vencer. O jogo consiste então em manter as escolhas feitas por Alice e Beto, fazer um número bem grande de rodadas, digamos N , e calcular a fração de vezes que Alice vence. Cabe a você escolher valores para N que lhe dê confiança sobre a resposta.

OBSERVAÇÃO: Note que valores maiores de N podem assegurar que flutuações estatísticas nas suas simulações não vão interferir no acerto de sua resposta.

Tome como exemplo as seguintes sequências de 3 bits escolhidas por Alice e Beto.

Alice: 000

Beto: 100

Note que, se em algum momento Casemiro sortear o bit '1', é garantido que Beto irá ganhar (pense sobre isso). Para que Alice ganhe, devem ser sorteados 3 bits '0' em sequência, e isso ocorre com probabilidade $1/2^3 = 1/8$. Logo, a sequência de Beto tem 7 vezes mais chances de ser a vencedora do que a sequência de Alice.

Tarefa 1: Dadas as sequências de Alice e Beto, estimar a probabilidade de Alice vencer o jogo.

Seguem alguns exemplos de entrada e saída. Em todos os exemplos a primeira linha contém a sequência de Alice, e a segunda linha contém a sequência de Beto. A saída deve ser apenas um número real correspondente à probabilidade estimada de Alice vencer. É garantido que ambas as sequências têm o mesmo comprimento (não precisa fazer verificação).

Exemplo de entrada 000 100	Saída para o exemplo de entrada 0.125124
---	--

Exemplo de entrada 001 011	Saída para o exemplo de entrada 0.666624
---	--

Exemplo de entrada 1111 0000	Saída para o exemplo de entrada 0.500005
---	--

Tarefa 2

Beto, após jogar diversas vezes o Jogo dos Bits contra Alice, cansou-se de perder. Nessa segunda parte do EP2 você terá que ajudar Beto a vencer Alice, ou determinar se isso é impossível. Alice, como já venceu Beto várias vezes, está extremamente confiante, a ponto de fornecer de antemão sua sequência escolhida.

Por exemplo, caso Alice escolha a sequência ‘010’, Beto pode escolher a sequência ‘001’, que é 2 vezes mais provável de ser a vencedora comparada com a sequência de Alice. (Se você duvida, faça um teste.)

Para ajudar Beto, você deve, dada a sequência de Alice, testar outras sequências (de mesmo comprimento) **até encontrar (se existente) uma que suas simulações sugerem que é melhor**. Aqui, para testar se uma sequência é melhor que outra, você precisa pensar bem a respeito, pois há casos de sequências que podem ter probabilidades semelhantes (sugerindo empate), e portanto não basta você testar se uma das probabilidades é maior que a outra. Por exemplo, você pode obter probabilidade 0.49995 para uma sequência de Alice e probabilidade 0.500005 para uma sequência de Beto; e isto não significa que a sequência de Beto é melhor.

Para testar outras sequências, você deve gerar —de uma maneira sistemática— as sequências, cuidando para que todas elas tenham o comprimento desejado. (Dica: trabalhar com números inteiros, fazer mudança de base, não se descuidar do comprimento da sequência, etc.)

Seguem alguns exemplos de entrada e saída. Em todos os exemplos, há uma única linha de entrada correspondente à sequência de Alice. A saída deve conter a sequência de Beto ou a string ‘Impossível’ (sem aspas), caso não haja uma melhor que a de Alice.

Exemplo de entrada 010	Saída para o exemplo de entrada 001
Exemplo de entrada 000	Saída para o exemplo de entrada 100
Exemplo de entrada 011	Saída para o exemplo de entrada 001
Exemplo de entrada 1111	Saída para o exemplo de entrada 0111

Instruções para entrega do EP

As instruções contidas aqui **devem ser rigorosamente seguidas**. Caso isso não ocorra, haverá **dedução da nota**. As instruções são as seguintes.

- 1) Seu envio no e-disciplinas deverá consistir de dois arquivos, com nomes “tarefa1.py” e “tarefa2.py”, um para cada tarefa enunciada acima.
- 2) O seu programa poderá usar somente os recursos da linguagem Python 3.x vistos até a última aula antes da entrega do EP. Se tiver dúvidas a respeito, pergunte ao professor de sua turma.
- 3) A entrada e a saída do programa devem ser **exatamente** como foi indicado nos exemplos, isto é, seu programa não deve imprimir nada além da saída e não deve ler nada além da entrada prevista. (Os valores não precisam ser exatamente os mesmos.)

Mensagens como “Digite a sequência” acarretarão redução da nota. (Isto tem a ver com a forma que os seus programas serão testados pelo monitor.) Quando você estiver fazendo/testando o seu programa, você pode ter tais mensagens, decidir como quer lidar com o número de jogadas N , mas na hora de preparar o arquivo para envio, você deve tirar ou comentar os comandos de impressão não solicitados.

- 4) Antes de entregar o seu exercício-programa, leia e siga atentamente as observações muito importantes contidas em **Instruções para a entrega de EPs em Python** (veja no e-disciplinas), onde estão descritas as instruções para a entrega dos exercícios-programas, os aspectos importantes na avaliação, a identificação no início do programa, etc.
- 5) O prazo limite para entrega deve ser obedecido rigorosamente. Programas fora do prazo não serão aceitos e receberão nota zero.
- 6) Se você tiver dúvidas, consulte o monitor ou o professor da turma.