Piggs and Bulls

Programação 1

Bernardo Marques 40535

Piggs and Bulls

Programação 1

Introdução

Neste trabalho é pedido para desenvolver em python um jogo chamado Pigs And Bulls. Este jogo é jogado por 2 jogadores, um pensa num número de 4 digitos de 0 a 9, sem repetição, e o outro tem que o adivinhar. Se a tentativa de adivinhar acertar um digito na posição certa então é um "touro", se acertar um digito, mas não for a posição certa então é um "porco".

O jogo será jogado entre 1 jogador e o computador, que gera um código aleatório. Para tal foi usado a função random() do modulo random.

Desenvolvimento

Para comparar ambos os códigos, o introduzido pelo jogador e o gerado aleatoriamente pelo computador, são transformardos em listas e uma função compara cada elemento de ambas as listas e verifica se é "touros" ou "porcos"

- transformar_lista() - esta função leva como argumento codigo (str), que será o código introduzido pelo jogador. Esta função vai iterar cada elemento da string e adiciona-lo como inteiro à lista cod_lista retornando-a.
- random_generator() – esta função cria uma lista de 4 elementos simulando um numero de 4 dígitos distintos entre 0 e 9. Para gerar tal lista criou-se a lista pool, que contem todos os números de 0 a 9, e de seguida dentro de uma iteração gera-se um número aleatório usando o método random() do módulo random. Visto que este método só gera um número no intervalo [0;1[, multiplica-se por 10 e converte-se para inteiro para obter um número inteiro entre 0 e 9 que é guardado na variável numero. Ao iterar, a função verifica se numero ultrapassa o comprimento da lista pool, daí a variável length (int). Caso número gerado aleatoriamente seja menor ou igual que o comprimento da lista, será utilizado como índice para anexar um número da lista pool à lista lrandom e removendo-o da anterior, que é a lista que se pretende; caso o número gerado seja superior ao comprimento da lista, então subtrai-se com o comprimento da lista. Como o comprimento da lista pool diminui e o numero gerado será sempre entre 0 e 9, então subtrai-se 1 à variável length. Quando a lista lrandom tiver os 4 elementos desejados a iteração acaba e devolve lrandom. (Nota: não foi usado len() pois sempre que a função iterava dava erro de ultrapassagem do alcance da lista)

- comparar() – esta função toma como argumentos lista2 (que será a lista gerada aleatoriamente, fora desta função) e tentativas (lista vazia) e que compara então as duas listas, a gerada aleatoriamente e a transformada da função transformar_lista(). Começa por pedir o código ao jogador e verificar se o código introduzido tem 4 digitos: se não, volta a invocar a função comparar(), se sim, invoca a função transformar_lista() para transformar o código numa lista. É introduzido 3 variáveis: n, touros e porcos, todas de valor 0 (int). n será usado para a contagem da iteração, esta que vai verificar se existe algum número que seja comum e que esteja na mesma posição em ambas as listas, caso sim adiciona 1 a touros, e vai verificar se tem um número em comum mas em posições diferentes, caso sim adiciona 1 a touros. Quando a iteração acaba, é introduzido a variável contagem de valor 0 (int), que será usada para a demonstração das tentativas feitas pelo jogador. A função então verifica o valor de touros, se for diferente de 4 então vai anexar à lista tentativas em string as variáveis codigo, touros e porcos, e vai verificar em vários casos se touros ou porcos têm valor 0, fazendo print do valor das variáveis para indicar ao jogador se acertou algum digito e soma 1 a contagem. Se touros for igual a 4 então a função soma 1 a contagem, anexa a tentativas as variáveis codigo, touros e porcos, faz print a indicar ao jogador que acertou e vai iterar pela lista tentativas e fazer print de cada elemento, demonstrando assim todos os códigos inseridos pelo jogador e o número de “touros e “porcos” de cada código inserido.