

## MAB225 – Computação II – Aula Prática 8

**Objetivo:** utilização de stdin, stdout e stderr

Atenção! Os testes da classe devem ser incluídos no mesmo arquivo “.py” em que a classe foi criada.

1. Crie uma classe que permite modificar a entrada da função input, a saída da função print e a saída de todos os erros:
  - a. Ao inicializar a classe, crie atributos pseudo-privados que guardam os valores originais de stdin, stdout e stderr.
  - b. Implemente o método setPrintFile. Esse método deve receber o nome de um arquivo como entrada (do tipo string), e modificar stdout de forma que a função print passe a imprimir no final do arquivo com nome igual ao passado por argumento.
  - c. Implemente o método resetPrint, que verifica se o nome do arquivo da variável padrão stdout é igual ao arquivo guardado no respectivo atributo privado. Em caso afirmativo, imprime uma mensagem informando que o arquivo já é o arquivo padrão e não pode ser fechado. Em caso negativo, fecha o arquivo que estava em stdout, modifica stdout para o valor guardado no atributo privado e imprime uma mensagem de sucesso.
  - d. Implemente o método setInputFile. Esse método deve receber o nome de um arquivo como entrada (do tipo string), verificar se o arquivo existe utilizando tratamento de exceção (FileNotFoundException é a exceção gerada caso o arquivo não exista), e, se existir, modificar stdin de forma que o comando input leia as linhas do arquivo dado. Se o arquivo não existir, imprima uma mensagem de erro.
  - e. Implemente o método resetInput, que verifica se o nome do arquivo da variável padrão stdin é igual ao arquivo guardado no respectivo atributo privado. Em caso afirmativo, imprime uma mensagem informando que o arquivo já é o arquivo padrão e não pode ser fechado. Em caso negativo, fecha o arquivo que estava em stdin, modifica stdin para o valor guardado no atributo privado e imprime uma mensagem de sucesso.
  - f. Implemente um método chamado de input. Esse método deve utilizar a função input para ler uma linha guardar seu valor em uma variável dentro de um tratamento de exceção. Se o arquivo que estiver sendo lido tiver chegado ao final a função input vai gerar a exceção EOFError. Se a exceção for gerada, retorne uma string vazia. Se não, retorne o valor guardado na variável.
  - g. Implemente o método setErrorFile. Esse método deve receber o nome de um arquivo como entrada (do tipo string), e modificar stderr de forma que os erros sejam impressos no final do arquivo com nome igual ao passado por argumento.
  - h. Implemente o método resetErrorFile, que verifica se o nome do arquivo da variável padrão stderr é igual ao arquivo guardado no respectivo atributo privado. Em caso afirmativo, imprime uma mensagem informando que o arquivo já é o arquivo padrão e não pode ser fechado. Em caso negativo, fecha o arquivo que estava em stderr, modifica stderr para o valor guardado no atributo privado e imprime uma mensagem de sucesso.
  - i. Teste a classe e verifique o que acontece com a função input, com a função print e com as mensagens de erro em cada um dos casos. Inclua os testes no arquivo “.py” onde a classe foi criada. Qual a diferença entre usar o método input da classe e a função input do python?