

Introdução

Nesta aula, vamos considerar o problema proposto no Gabarito 4 da P08 da semana passada. Vamos melhorar a solução passando informações relevantes à execução do programa por meio da linha de comandos do *shell*. Infelizmente, a nossa IDE IDLE não tem recursos apropriados para fazer isto. Logo vamos ter de utilizar um emulador de terminal nesta aula. **Observação:** Há uma sugestão de solução do exercício da aula passada na seção de Materiais no sistema de entrega de trabalhos.

Problema a ser resolvido

Escreva um programa que leia os dados dos alunos de uma disciplina contendo a seguinte informação sobre cada aluno:

- Número de matrícula (*matr*)
- Nome do aluno (*nome*)
- Curso do aluno (*curso*)
- Nota final (*nota*)
- Número de faltas (*faltas*)

Os dados já estão digitados em um arquivo previamente preparado em que cada linha corresponde a um aluno. Os itens estão separados por vírgula em cada linha. Por exemplo, a linha:

87543,Ivo de Castro,Agronomia,67,14

refere-se ao aluno Ivo de Castro, de matrícula 87543. Ele faz o curso de Agronomia, obteve 67 pontos e teve 14 faltas na disciplina.

Além da leitura, os dados devem ser incorporados a uma lista de 5-tuplas. Cada tupla corresponde a um aluno. Como resultado, o programa deve emitir uma listagem contendo os alunos que foram reprovados por faltas ou por nota, supondo que a disciplina tenha um total de n horas-aulas de carga horária semestral a ser obtida dos parâmetros da linha de comando de execução do programa. Além disso, um aluno para conseguir aprovação tem de estar presente a, pelo menos, 75% das aulas. Por nota, o aluno tem de ter nota final de, no mínimo, 60 pontos.

Instruções

1. Abra o IDLE e crie um novo arquivo fonte denominado *p09.py*. Não se esqueça de salvá-lo de tempos em tempos, porque pode ocorrer falha de energia elétrica durante a aula prática.
2. Digite os comentários obrigatórios (*nome*, *matrícula*, *data* e uma breve descrição sobre o que o programa faz).
3. Estruture seu programa em três funções pelo menos: *main()*, *leiaAlunos(nomearq)* e *verifiqueReprovados(bd)*.
4. A função *main()* deve fazer o seguinte: usar a variável *sys.argv* da biblioteca *sys* para manipular os parâmetros passados pela linha de comandos, chamar a função *leiaAlunos(nomearq)*, depois a *verifiqueReprovados(bd)* que imprime o relatório pedido. Os parâmetros a ser passados pela linha de comando são, além do próprio nome do programa fonte obviamente: nome da disciplina, carga horária total da disciplina no semestre e o nome do arquivo com os dados dos alunos. Se, por acaso, o usuário não passar todos os

Prática 9 – INF101 – 2018/II – 2 pontos

parâmetros, a função `main` deve emitir uma mensagem de erro apropriada. Veja o último exemplo de execução do programa abaixo.

5. Implemente a função `leiaAlunos(nomearq)` com um parâmetro contendo o nome do arquivo a ser lido e ela retorna a lista contendo o banco de dados.
6. Implemente a função `verifiqueReprovados(bd)` com um parâmetro: o banco de dados. Ela varre todo o banco de dados `bd` e imprime os dados dos alunos que se reprovaram por faltas ou por nota. Veja, nos exemplos abaixo, resultados de execuções do programa.
7. Não se esqueça também de chamar a função `main` no final de seu código fonte para iniciar todo o processo.
8. Para rodar o programa, não use o mecanismo de execução do IDLE, porque ele não permite passar parâmetros diretamente pela linha de comando. Para tanto, abra um emulador de terminal do sistema por meio do atalho do teclado: CTRL-ALT-T. Ou por meio de algum menu ou ícone da interface gráfica. Se não conseguir, peça ajuda ao professor.
9. Teste seu programa com os arquivos de dados fornecidos pelo sistema de entrega de trabalhos: `let374.dat` e `nut350.txt`.
10. Se seu programa entrar em *laço infinito*, tecla CTRL-C na janela do *Python Shell* para interromper a execução do programa.

Após certificar-se de que seu programa esteja correto, envie o arquivo do programa fonte (`p09.py`) através do sistema de entrega do LBI. Abaixo estão exemplos de execução do programa usando os dois arquivos de entrada fornecidos no sistema de entrega de trabalhos.

Exemplos de execução do programa

Execução a partir da seguinte linha de comando no *prompt* do terminal:

```
$ python3 p09.py LET374 60 let374.dat
```

Alunos Reprovados na Disciplina LET374				
Matrícula	Nome	Curso	Nota	Faltas
87680	Ana Lopes	Secretariado	22	18
89299	Gregório Lima	Matemática	74	16
89327	Lupicínio Rocha	Física	56	2
89401	Ricardo Faria Teixeira	Química	48	18
90012	Luís Graham	Engenharia Mecânica	40	15
90028	Carlos Costa	Engenharia Civil	72	25

Execução a partir da seguinte linha de comando no *prompt* do terminal:

```
$ python3 p09.py "NUT 350" 45 nut350.txt
```

Alunos Reprovados na Disciplina NUT 350

Matrícula	Nome	Curso	Nota	Faltas
79824	Salete Sales	Nutrição	78	12
80013	Fiona Malaquias	Nutrição	65	20
85910	Penélope Poubel	Enfermagem	56	15
85920	Agamenon Lopes	Nutrição	61	18

Execução a partir da seguinte linha de comando no *prompt* do terminal:

```
$ python3 p09.py
```

```
***ERRO: Este programa deve ser executado com três argumentos:  
python3 p09.py <nome da disciplina> <carga horária> <arquivo de dados>
```