

Recursão e Ordenação

O professor Hilario, de idade já avançada, leciona a disciplina “Programação 2” há algumas décadas e já teve centenas de milhares de alunos. Ele sempre aplica 1 prova valendo 30 pontos, 1 simulação da Maratona de Programação valendo 30 pontos e 1 trabalho prático valendo 40. Curioso, Hilario decidiu fazer algumas estatísticas a respeito das notas ao longo das várias décadas em que ele dá aula. Ele armazena uma planilha com todas as notas dos alunos desde os primórdios. A planilha contém:

- A matrícula do aluno.
- O nome do aluno.
- O semestre letivo em que o aluno cursou a disciplina.
- A nota do aluno em cada uma das 3 avaliações.
- Bônus para cada vez que um aluno vai ao quadro resolver um exercício.
- O número de faltas do aluno.

A **nota total** do aluno é a soma das notas das 3 avaliações, acrescida dos possíveis bônus. Além do bônus por resolver exercícios no quadro, há ainda duas outras formas de conseguir pontos extras:

1. Caso o aluno não tenha faltado nenhuma aula, ele ganha 2 pontos de bônus.
2. Caso seu trabalho esteja correto e esteja entre os mais rápidos da turma, sua nota será acrescida de 2 pontos.

Apesar de tantos pontos extras, a **nota final** do aluno não pode ultrapassar 100 pontos, mesmo que a **nota total**, incluindo os bônus, seja maior que 100.

Com preguiça calcular as tais estatísticas, Hilario resolveu passar isso como o trabalho prático desse semestre. Sua missão será a seguinte:

1. Leia o arquivo binário “entrada.bin” fornecido pelo professor, contendo a planilha de notas no formato de um dicionário em Python. A chave do dicionário é a matrícula do aluno, e o conteúdo é uma tupla contendo o nome, o semestre letivo (tupla com dois inteiros representando o ano e o semestre), as notas do aluno (tupla com 4 inteiros representando, respectivamente, a nota da primeira prova, a nota do trabalho e a nota da Maratona e o bônus de exercícios) e um inteiro com o número de faltas. Considere que o arquivo binário foi criada através da biblioteca ***pickle***.
2. Crie uma lista contendo **APENAS** os números de matrícula dos alunos. Ordene **essa lista** de acordo com os seguintes critérios, nesta ordem:

- (a) Pelo semestre letivo (do mais recente para o mais antigo).
- (b) Em caso de empate, pela **nota final** (da maior para a menor).
- (c) Em caso de novo empate, em ordem alfabética do nome (os caracteres são ordenados de acordo com suas posições na tabela UTF-8).
- (d) Caso ainda haja empate, em ordem crescente de matrícula (não existem duas matrículas iguais).

A ordenação deve utilizar uma única função de comparação que recebe duas matrículas m_1 e m_2 , além do dicionário de alunos, e retorna:

- Verdadeiro, caso o aluno da matrícula m_1 deva aparecer na lista ordenada **antes** do aluno da matrícula m_2 .
 - Falso, caso contrário.
3. Crie um arquivo texto chamado “saida.txt” contendo o resultado da ordenação. Cada linha do arquivo contém os dados de um aluno, seguindo a ordem desejada pelo professor. Para cada aluno, imprima:
- O semestre letivo. Ex: 2023/1
 - Matrícula
 - Nome
 - Nota final (que não ultrapassa 100)
 - Somatória de cada nota e possíveis bônus, resultando na notal total.

O formato de saída deve **EXATAMENTE** como no exemplo a seguir:

```
2023/1 BSI0381767 Ana Ferreira - 100 (29+40+29 +2P = 100)
2023/1 BSI0791174 João Jesus - 100 (28+42+30 +2P = 102)
2023/1 BSI0473077 Maria Albuquerque - 100 (27+42+30 +1E +2P = 102)
2023/1 BSI1858432 Afonso Coimbra - 98 (29+42+26 +1E = 98)
2023/1 BSI0878237 Ana Jesus - 98 (30+37+29 +2E = 98)
2023/1 BSI1718106 Beatriz Gomes - 87 (30+33+23 +1E = 87)
2023/1 BSI1720783 Francisco Correia - 87 (29+37+21 = 87)
2023/1 BSI1243598 Rodrigo Almeida - 57 (16+34+5 +2P = 57)
2023/1 BSI1631563 Tiago Silva - 13 (4+0+7 +2P = 13)
2022/2 BSI0528369 Afonso Teixeira - 100 (29+40+30 +1E = 100)
2022/2 BSI1670525 Afonso Castro - 94 (29+38+25 +2P = 94)
2022/2 BSI0706526 Miguel Vaz - 94 (29+37+28 = 94)
2022/2 BSI0763661 Martim Alves - 81 (24+36+21 = 81)
2022/2 BSI0811452 Martim Alves - 81 (18+38+25 = 81)
2022/2 BSI0726473 Francisca Sousa - 42 (24+5+12 +1E = 42)
2022/2 BSI0676903 Madalena Fernandes - 42 (13+8+21 = 42)
```

Note que Ana, João e Maria empataram na nota total em 2023/1, apesar de João e Maria terem ultrapassado os 100 pontos com os bônus. João conseguiu 2 pontos de bônus no trabalho (tirou 42) e também obteve 2 pontos extras por presença (identificado na saída por +2P). Já Maria também esteve entre os trabalhos mais rápidos (42 pontos), obteve 1 ponto extra com os exercícios (+1E) e ainda conseguiu 2 pontos por presença (+2P).

A ordenação deve ser feita com algum dos algoritmos recursivos de ordenação ensinados nesta disciplina que executem em $O(n \cdot \lg n)$ no caso médio, e deve ser executada uma única vez (ou seja, uma única ordenação já deve ser capaz de classificar os alunos de acordo com todos os critérios citados acima). **Atenção:** não imprima nada na tela. Absolutamente nada! E não salve nada no arquivo texto além do que foi pedido.

Bibliotecas

O arquivo binário foi criado através da biblioteca ***pickle***, portanto a mesma biblioteca deve ser utilizada para fazer a leitura do dicionário com os dados de entrada. **Nenhuma** outra biblioteca pode ser utilizada, a não ser que tenha sido implementada pelo próprio aluno e seu código fonte seja entregue junto do restante do código.

Observações

- O trabalho vale 40 pontos, pode ser feito em grupo de até 3 integrantes.
- Trabalhos considerados **plágio** terão nota 0 para quem copiou e para quem forneceu o trabalho. Além disso, serão enviados para o Conselho de Ética.
- O código deve ser feito em Python3. Deve ser enviado um arquivo compactado contendo **apenas** os códigos fontes (*.py) do programa. Não inclua arquivos de entrada e/ou de saída na submissão do trabalho.
- Trabalhos entregues após o prazo, com erro de execução, com formato de saída incorreto, ou que não compilarem terão nota 0.
- O trabalho deve ser enviado na sala da disciplina do AVA.
- Em caso de dúvidas na especificação do trabalho ou no próprio trabalho, contate-me em hsjunioe@gmail.com