

Projeto prático 2

SCC 0224 – Estrutura de Dados II
Prof. Maria Cristina Ferreira de Oliveira
16 de abril de 2024

1. Descrição

O ICMC está selecionando alunos para bolsas de intercâmbio, mas há um problema: a quantidade de bolsas é limitada! Para garantir uma seleção justa, serão selecionados os candidatos com as melhores médias ponderadas e menor quantidade de reprovações. Confiando nos seus conhecimentos sobre algoritmos de ordenação, o ICMC contratou você para elaborar um programa para fazer a seleção dos alunos que receberão bolsa.

O critério de seleção será, primeiramente, pela maior média, e depois pelo menor número de reprovações. No caso de dois ou mais candidatos terem exatamente a mesma média e número de reprovações e estarem no limite do número de vagas, ambos recebem a bolsa, entretanto, bolsas não serão alocadas para alunos com mais de 10 reprovações.

Você vai utilizar os seguintes algoritmos de ordenação: T = 1: Insertion sort; T = 2: Heapsort

2. Entrada e Saída

O programa deverá receber primeiramente T, o método de ordenação a ser utilizado para selecionar os alunos. A próxima linha contém N e M, respectivamente o número de alunos que aplicaram para a bolsa de intercâmbio, e o número de bolsas disponíveis. As próximas N linhas deverão receber a_i , b_i e c_i , respectivamente a média ponderada ($0.00 \leq a_i \leq 10.00$), o número de reprovações ($0 \leq b_i \leq 20$) e o nome do aluno.

Como saída, seu programa deve imprimir X, o número de alunos que receberão a bolsa de intercâmbio. Lembre-se que poderá haver poucas inscrições, ou vários alunos com a mesma média e número de reprovações. E nas próximas X linhas deve imprimir, em ordem de prioridade, os nomes dos alunos selecionados para a bolsa. Nos casos de empate, os alunos devem ser listados em ordem alfabética.

3. Conteúdo e data de entrega

O trabalho é individual e a data limite de submissão é 12/05/2024 (trabalhos em atraso não serão aceitos), no sistema RunCodes (<https://runcodes.icmc.usp.br/>). O formato da entrega deve ser um arquivo com o algoritmo de ordenação para os casos de entrada disponíveis.

- O código deverá conter comentário/documentação adequados;
- O trabalho pode ser desenvolvido utilizando C ou Python, a escolha;
- Não é permitido utilizar bibliotecas que já disponibilizam os algoritmos de forma parcial ou total;
- Todo o código do projeto deve ser desenvolvido pelo próprio aluno. Os códigos serão submetidos a um detector de plágio, e a utilização de código criado por terceiros será tratada como plágio (nota zero).

4. Critérios de avaliação

- 9,0 pontos pela implementação dos algoritmos solicitados, de forma correta (avaliado segundo o resultado do RunCodes, e por inspeção do código).
- 1,0 ponto por código legível e bem comentado.

5. Exemplos de entrada e saída

Entrada	Saída
1 3 3 9.05 0 Rafael 5.54 0 Joao 6.65 0 Carlos	3 Rafael Carlos Joao

Entrada	Saída
1 3 2 9.00 1 Pedro 10.00 0 Ana 8.50 3 Alan	2 Ana Pedro

Entrada	Saída
2 5 2 7.50 0 Rafael 8.20 1 Ana 5.55 2 Carlos 4.08 4 Vitoria 6.60 2 Marcos	2 Ana Rafael

Entrada	Saída
2 3 4 7.05 3 Mario 7.05 0 Rodrigo 5.00 3 Beatriz	3 Rodrigo Mario Beatriz

Entrada	Saída
1 7 4 9.00 0 Pedro 9.00 0 Ana 8.50 2 Alan 7.05 1 Rodrigo 6.58 3 Vitor 5.59 4 Beatriz 7.05 1 Amanda	5 Ana Pedro Alan Amanda Rodrigo

Note que independente do método de ordenação utilizado a saída será sempre a mesma.