SISTEMA DE RANKING PARA MARATONA DE PROGRAMAÇÃO DA UEFS

Felipe Gomes da Silva

Engenharia da Computação - Universidade Estadual de Feira de Santana (UEFS)
Feira de Santana – BA, Brasil – 44036-900

felipegoomes8910@gmail.com

Abstract. *UEFS is organizing a Programming Marathon for students studying Computer Engineering, Information Systems and Computer Science and requested a system to manage team scores, calculating them and updating the overall ranking. This article aims to describe the entire process of creating this system with the Python language.*

Resumo. A *UEFS está organizando uma Maratona de Programação para os alunos dos cursos de Engenharia da Computação, Sistemas de Informação e Ciência da Computação e solicitou um sistema para gerenciar a pontuação das equipes, calculando esta e atualizando o ranking geral. Esse artigo tem por objetivo descrever como foi todo o processo de criação desse sistema com a linguagem Python.*

1.Introdução

Um sistema de ranking é utilizado para estabelecer a classificação em tempo real de equipes ou atletas em uma competição ou modalidade. Para isso, é necessária a criação de um programa que analise os dados e faça o processamento destes para assim ser exibido o Ranking completo com nome da equipe, quantidade de questões resolvidas separadas por tipo (fácil, média e difícil) e o tempo levado para resolução; em caso de empate, os critérios considerados foram os de quantidade de questões difíceis e tempo que foram levados a uso. Portanto, esse artigo fornece as informações necessárias do programa feito em Python para gerar o que foi solicitado e assim a conclusão do PBL.

Diante disso, a proposta desse relatório é apresentar as fases de criação do sistema e a maneira como ele computa os dados. Ao decorrer dos tópicos será apresentada a estrutura do código feito na linguagem do Python e como ele gera os dados necessários de forma eficiente e rápida para o usuário.

2. Metodologia

2.1 Desenvolvimento nas sessões

Para criação do projeto foi necessário fazer sessões com a turma de MI - Algoritmos. Diante da leitura do problema questões foram levantadas: 1) Como fazer os critérios de desempate; 2) O que é uma média de soma por equipe? 3) Como fazer o fluxograma solicitado.

O problema inicial foi majoritariamente a falta de conhecimento da turma, poucos sabiam o básico do Python, por esse motivo, uma das metas estabelecidas foi pesquisar e aplicar o conhecimento absorvido para iniciar o projeto do programa. Nas sessões seguintes, totalizando 4 sessões, foi discutido como poderia ser feito um ranking funcional, pensando em qual unidade de tempo seria usada e como as condicionais *if, elif e else* poderiam ser usadas nos sistemas. Além disso, surgiram dúvidas sobre funções prontas (como listas e *max*) para os critérios de desempate e de como deixar o código organizado em poucas linhas.

Foi definido que não era necessário senha e nem outras coisas que fossem afetar o código futuramente, apenas o básico poderia ser usado. Seguindo para a finalização do projeto, foi solicitada a criação de um menu para o usuário, no qual, ao selecionar um caractere específico, fosse possível adicionar as informações de cada equipe e finalizar com determinado caractere, aparecendo a tabela e o ranking completo e então encerrando o programa.

2.2 Definição de requisitos/funcionalidades

O passo inicial da criação do código foi inserir a variável de saída print, para mostrar ao usuário que ele deve personalizar o valor de cada questão e os nomes das equipes que participarão da Maratona de Programação. O processo de adicionar o peso das questões (fáceis, médias, difíceis) e a definição dos nomes das equipes é feito por meio das variáveis de entrada input. A fim de conseguir gerar o ranking e exibir suas devidas pontuações na tabela das equipes, foi necessário definir variáveis igualadas a ' ' (vazio), para que houvesse a mudança de valores, para que ficasse livre para alterações de processamento, diminuindo ou aumentando seu valor, sem que afetasse o funcionamento do código.

Após isso, foi criada a estrutura de repetição While, para que o usuário escolha uma dentre seis opções desejadas e selecione de 1 a 5 para as equipes, finalizando, então, o looping digitando 6. Quando o usuário digitar algum dos cinco primeiros números citados, ele poderá inserir a quantidade de questões acertadas por categoria de dificuldade e o tempo gasto para resolução, assim, automaticamente o sistema fará o cálculo da pontuação da equipe com base nos acertos, armazenará as questões difíceis acertadas e o tempo utilizado para o critério de desempate.

Dentro do laço de repetição existem condicionais auxiliares que definem a colocação na Tabela de Maratona. De acordo com cada pontuação de cada equipe, a variável *rankx* (x representa o número da equipe) localizada dentro de cada bloco de *if*, é responsável por essa mudança e pela atualização do ranking da tabela.

Quando o usuário digitar 6 o looping será finalizado, assim como dito anteriormente, e, após isso, a variável print exibirá a Tabela de ranking completa, contendo: equipes e suas pontuações, problemas resolvidos por categoria, tempo gasto para resolução dos problemas, média geral, após a exibição da tabela virá outra sequência de *print* que mostrará a equipe vencedora e sua pontuação, quantas questões difíceis ela resolveu o tempo gasto para resolução de todos os problemas. Juntamente a isso, será exibida a equipe que resolveu mais questões difíceis e a quantia respectiva.

2.3 Ordem de Codificação

A ordem de codificação foi simples e foi dividida em 4 partes:

1 - A primeira parte inicia com as variáveis *input*, para personalização do valor das questões e nomes das equipes.

2 - A segunda parte contém variáveis para armazenar, processar os dados e gerar o ranking.

3 - A terceira parte é formada pela estrutura de repetição While, na qual é feito o looping. Lá existem variáveis auxiliares e blocos de *if, elif e else*, para haver a atualização do ranking da tabela, definição da equipe com mais acertos difíceis e a equipe campeã da maratona; após isso é mostrado o cálculo da média de pontos geral.

4 - Na quarta parte estão localizadas as variáveis de saída *print*, que irão exibir a Tabela de Ranking das Equipes, a média de pontos, a equipe vencedora e a com mais acertos dificeis.

3.Resultados e Discussões

As dificuldades encontradas durante as sessões explicativas para a execução do ranking foram facilmente resolvidas com a ajuda mútua e, principalmente, com as dicas do tutor durante a sessão, que explicou uma lógica que aplicada ao Python que funcionou perfeitamente. Os critérios de desempate utilizaram uma lógica parecida com a do ranking, assim, quando a definição do ranking estava montada, grande parte do código foi definido, necessitando apenas de pequenos ajustes e organização da lógica de *or, and* e </> para os critérios de desempate.

Grande parte das ideias surgiram após as sessões com a turma de MI - Algoritmos, dentre elas estão os laços de repetição e o uso do *if, elif* e else para comparar e definir a colocação de cada equipe.

3.1. Manual de uso/entradas e saídas

Quando iniciado, é necessário definir o peso de cada questão e os nomes das equipes que irão participar da Maratona. Em seguida, o menu aparecerá e o usuário deve selecionar

qual equipe deseja inserir os dados, assim como exemplificado na Figure 1, para perfeita execução do sistema é necessário que todas as equipes estejam com suas devidas informações preenchidas corretamente.

Depois da inserção de cada equipe de forma unitária, a tecla 6 deve ser apertada para que o código seja finalizado e todos os dados sejam processados, e assim a Tabela do Ranking poderá ser apresentada de forma organizada, seguida com o destaque da equipe vencedora, seus dados e, juntamente, exibir a equipe com mais acertos de questões dificeis, assim como na Figure 2.

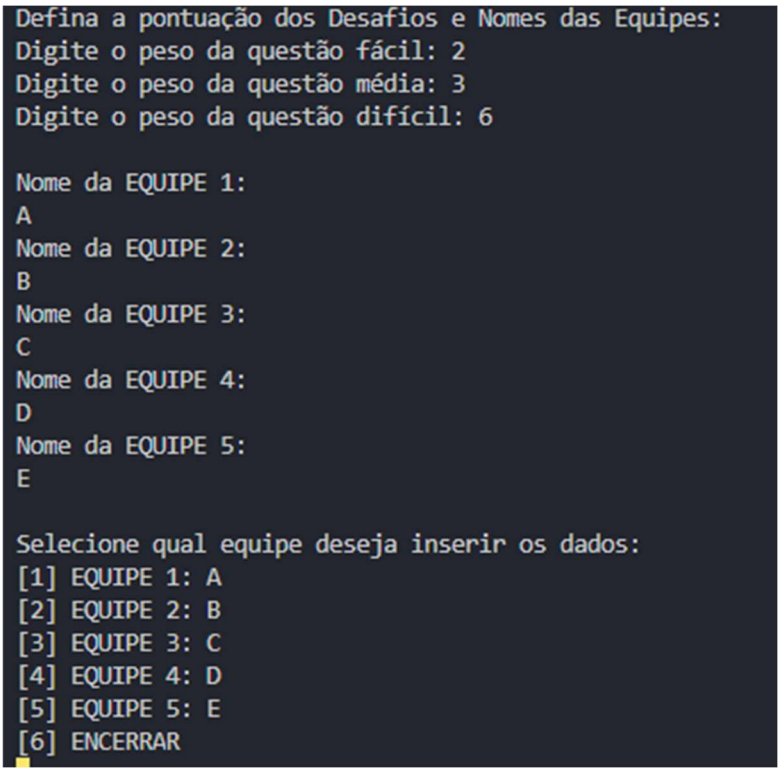

Figura 1. Início do Programa e Menu.

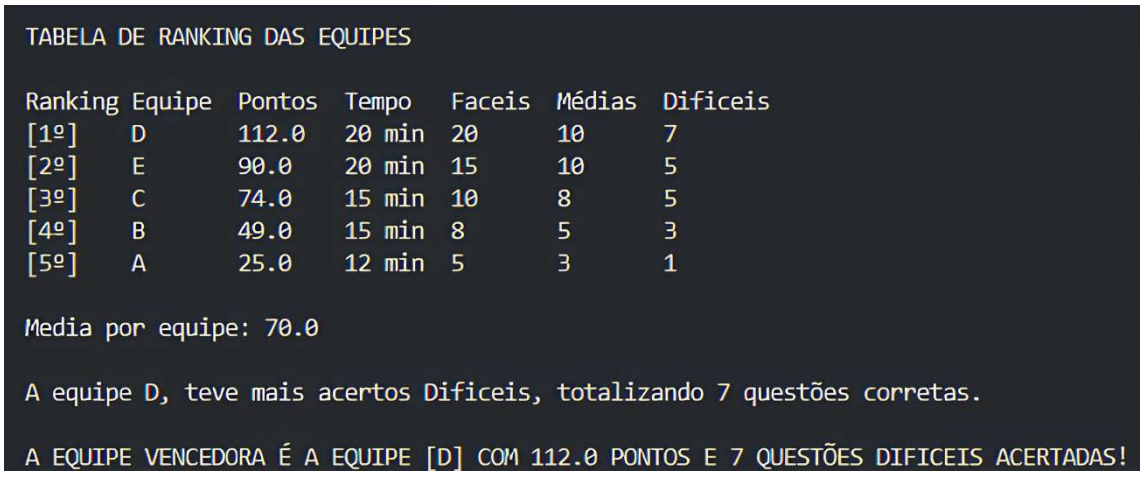

Figura 2. Tabela de Ranking das Equipes e finalização do programa.

3.2. Testes e Erros

O programa foi criado na versão 3.12.5, como sugerido nas sessões, com o sistema operacional do *Windows 10*. Durante os testes houveram erros de lógica que faziam o código não funcionar ou inverter as posições, exibir pontuações sem precisão, o que foi resolvido analisando o próprio código e buscando conhecimento nas sessões de MI-Algoritmos.

Para resolver bugs, foi necessário observar a sintaxe do código, o incremento de blocos de if, elif e else além da definição de novas variáveis para organização e exibição do rank.

4. Conclusão

O relatório apresentou a construção e a maneira de utilizar o *software* criado para gerar a Tabela de Ranking da Maratona de Programação da UEFS e nota-se que o problema apresentado foi resolvido e concluído em Python. O sistema apresenta os seis tópicos solicitados e pode ser reutilizado futuramente para outras competições, deixando a critério do usuário o peso de cada problema a ser resolvido, sem qualquer limitação.

Há a possibilidade de apresentação de melhorias de acordo com as atualizações da IDE, sendo plausível a adição de mais categorias caso o usuário precise. Ademais, tendo em vista que o código não possui tratamento de erro de digitação, é possível que haja problemas caso algo errado seja inserido, fazendo com que o usuário seja submetido à refação de todo o processo de inserção de dados.

5. Referências Bibliográficas

Guanabara. (Curso em Vídeo), Curso em Python\#10 - Condições (parte 1). Youtube,(2017). Disponível em: <https://youtu.be/K10u3XIf1-Q?si=NZknxtdb6KddUli4>. Acesso em 05 de set de 2024.

Guanabara. (Curso em Vídeo), Curso em Python\#12 - Condições Aninhadas. Youtube (2017). Disponível em: <https://youtu.be/j9bYDjaAYzw?si=zyFnbyd9eWujOmV0>. Acesso em 10 de set de 2024.

Guanabara. (Curso em Vídeo), Curso em Python\#14 - Estrutura de repetição while. Youtube, (2017). Disponível em: <https://youtu.be/LH6OIn2lBaI?si=zHIiesFYPEysM6Z1>. Acesso em 10 de set de 2024.

SANTOS, Leonardo. O que é: Ranking Global? SempreTopGames, (2024).0 Disponível em: <https://sempretopgames.com.br/glossario/o-que-e-ranking-global-como-funciona-e-importancia/>. Acesso em: 13 de set de 2024.

SIGNIFICADO de Ranking. Dicio. Disponível em: <https://www.dicio.com.br/ranking/#:~:text=Significado%20de%20Ranking,posi%C3%A7%C3%B5es%20em%20ranking%20de%20universidades>. Acesso em: 15 de set de 2024.