

Atividade Prática 11

Aluno: Lucas de Oliveira Araújo

Matrícula: 2022036039

Introdução

Este relatório tem como objetivo explorar o uso da estrutura de dados de conjuntos disjuntos na determinação de classes de equivalência para uma relação de equivalência. Uma relação de equivalência é caracterizada por ser reflexiva, simétrica e transitiva. Neste contexto, iremos considerar um exemplo específico de relação de equivalência, que é o resto da divisão por um inteiro fixo 'p'.

Para demonstrar a utilidade da estrutura de conjuntos disjuntos nesse contexto, utilizaremos um TAD (Tipo Abstrato de Dados) fornecido pelo professor. Esse TAD representa uma relação e possui operações para criar uma nova relação definida nos inteiros de 1 até 'n', bem como determinar se um par ordenado pertence ou não à relação.

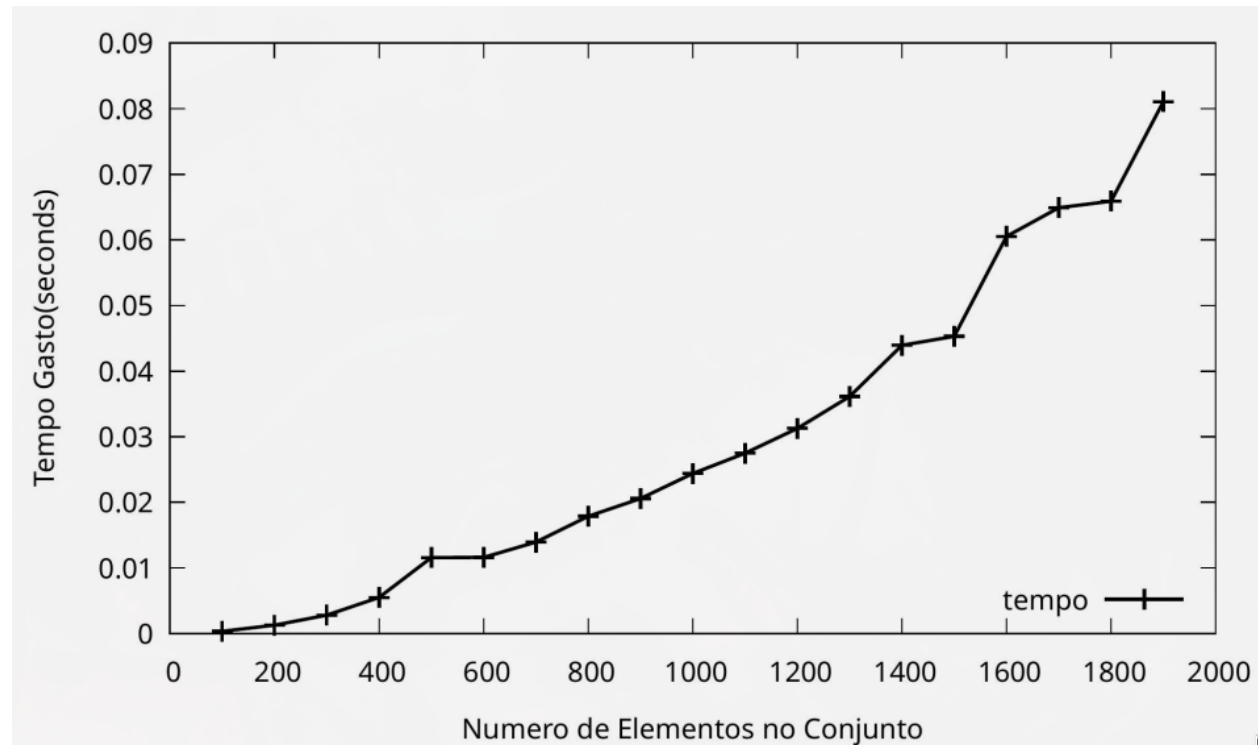
Ao considerar a relação de equivalência baseada no resto da divisão por 'p', podemos observar propriedades interessantes. Por exemplo, se dois números inteiros, 'x' e 'y', têm o mesmo resto da divisão por 'p', então eles pertencem à mesma classe de equivalência. Dessa forma, podemos agrupar elementos em conjuntos disjuntos, onde cada conjunto representa uma classe de equivalência.

A estrutura de conjuntos disjuntos nos permite criar e manipular conjuntos, além de realizar operações como unir conjuntos e verificar se dois elementos pertencem ao mesmo conjunto. Essas operações serão fundamentais para determinar as classes de equivalência a partir da relação definida.

Neste relatório, iremos analisar os dados fornecidos, que consistem em pares ordenados representando elementos e sua pertinência à relação de equivalência. Utilizaremos a estrutura de conjuntos disjuntos para agrupar os elementos em suas respectivas classes de equivalência. Além disso, examinaremos o tempo de execução necessário para realizar essa tarefa, a fim de avaliar a eficiência do algoritmo utilizado.

Testes realizados

A bateria de testes foi realizada variando o número de elementos no conjunto em 100 até um total de 2 mil elementos por conjunto. Os dados são expostos no gráfico abaixo.



Ao observar os dados, podemos notar que à medida que o tamanho do conjunto aumenta, o tempo de execução também aumenta. Isso indica uma relação direta entre o número de elementos no conjunto disjunto e o tempo necessário para executar o algoritmo. Além disso, podemos observar que o tempo de execução aumenta de forma acentuada à medida que o número de elementos no conjunto disjunto aumenta. Isso sugere que o algoritmo utilizado tem uma complexidade de tempo que cresce de forma mais rápida à medida que o tamanho do conjunto aumenta.

Embora haja uma tendência geral de aumento no tempo de execução, existem algumas variações nos valores. Por exemplo, para um conjunto de 600 elementos, o tempo de execução é muito semelhante ao do conjunto de 500 elementos. Isso pode ser devido a fatores como o hardware em que o algoritmo foi executado, a eficiência da implementação do algoritmo ou a características específicas dos dados de entrada.

Podemos observar que em alguns pontos específicos, como para conjuntos de 1000 e 1900 elementos, há saltos significativos no tempo de execução. Isso pode indicar que esses pontos

são pontos críticos para o algoritmo ou que existem operações ou etapas adicionais sendo executadas para esses tamanhos específicos de conjunto.

Conclusão

Com base na análise dos dados fornecidos, podemos concluir que o tempo de execução do algoritmo de conjunto disjunto aumenta à medida que o número de elementos no conjunto aumenta. A taxa de crescimento do tempo de execução parece ser aproximadamente exponencial, indicando que o algoritmo tem uma complexidade de tempo que aumenta rapidamente à medida que o tamanho do conjunto cresce. No entanto, também foram observadas variações nos valores e saltos significativos em alguns pontos, sugerindo que outros fatores além do tamanho do conjunto podem influenciar o tempo de execução.