Relatório 2º Projeto ASA 2019/2020

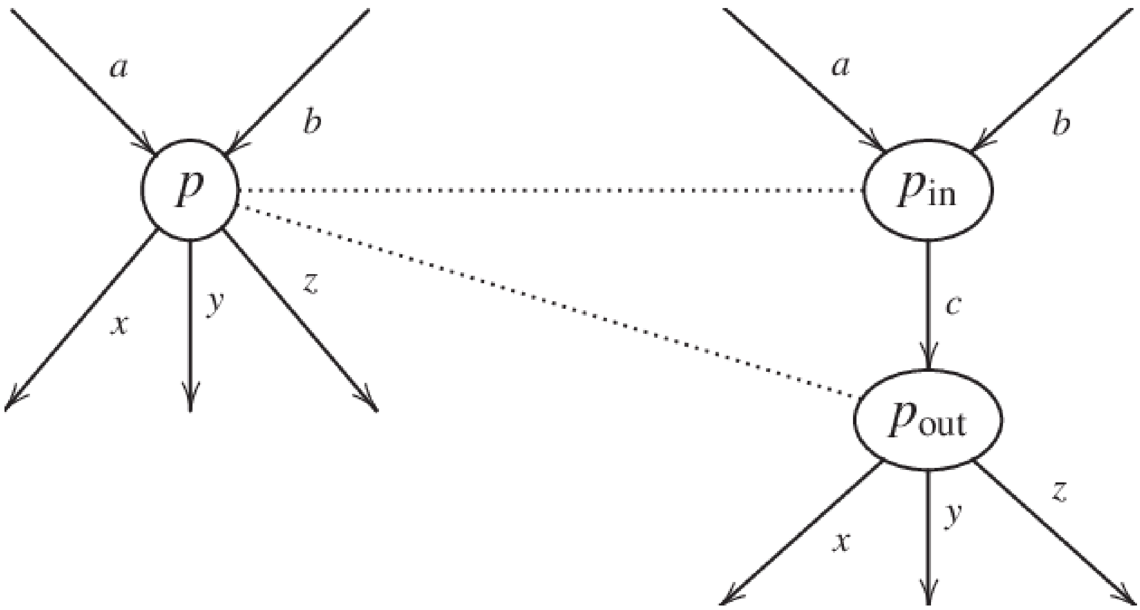
Grupo: al102

Aluno(s): Diogo Barbosa (92451) e Rafael Gonçalves (92544)

Descrição do Problema e da Solução:

Com o objetivo de minimizar ao máximo encontros entre pessoas numa cidade para prevenção do contágio de um vírus, foi pedida a elaboração de um programa que, conhecendo a distribuição das ruas e avenidas da cidade (quadriculado regular), os cruzamentos entre ruas e avenidas onde estão localizados os supermercados e as casas dos habitantes que pretendem sair, fornecesse o número máximo de pessoas que poderiam deslocar-se a um supermercado com probabilidade nula de se encontrarem com outra pessoa, ou seja, cada pessoa teria de ir por um caminho para um supermercado que não intercetasse outro caminho qualquer de outra pessoa.

Para encontrar uma solução, o grupo decidiu pensar na situação como um problema de fluxo máximo. Da maneira como surge explicitado no enunciado, a representação natural da situação é como um grafo não-direcionado em que os cruzamentos são vértices com capacidade 1 e as estradas que ligam cruzamentos são arcos com capacidade 1. No entanto, para o problema ser passível de resolução com algoritmos de fluxo máximo, são necessárias algumas alterações: primeiro, o grafo tem de ser transformado num grafo dirigido, pelo que duplicamos cada aresta de modo a que para uma aresta (A, B) seja criado um arco de A para B e um arco de B para A, ambos com capacidade 1; segundo, para simular a capacidade dos vértices, duplicam-se os vértices, em que um será o vértice in, onde se ligam todos os arcos que incidiam no original, e um vértice out, que liga a todos os vértices a que o original se ligava. O vértice in liga-se ao vértice out com capacidade igual à capacidade do vértice original (1). As capacidades dos restantes arcos mantêm-se a 1. A figura em baixo clarifica a transformação dos vértices originais.



Relatório 2º Projeto ASA 2019/2020

Grupo: al102

Aluno(s): Diogo Barbosa (92451) e Rafael Gonçalves (92544)

Criamos um vértice fonte e um vértice destino separados do grafo. Em seguida, ligamos a fonte a todos os vértices com casas de pessoas, e todos os vértices com supermercados ligam-se ao destino.

Por fim, corremos o algoritmo de Edmonds-Karp para encontrar o fluxo máximo entre a fonte e o destino.

Análise Teórica:

Em pseudo-código muito simplificado, o programa realiza as seguintes operações:

Graph g;
ReadInput();
InitializeGraph(g);
EdmondsKarp(g);

A complexidade será assim:

- Leitura dos dados de entrada: O(S + C) pois lemos todos os pares (x,y) correspondentes a supermercados (S) e habitações (C).
- Criação do grafo: O(N * M) = O(V + E) pois inicializamos todos os vértices (2 * N * M + 2) e arcos(2 * (2 * (M – 1) * N + 2 * (N – 1) * M + S + C)).
- Aplicação do algoritmo Edmonds-Karp: O(V^2) pois realizamos min(S,C) buscas em largura (O(V + E)), o valor máximo de min(S,C) = V, logo a complexidade é O(V + E), como E é proporcional a V, podemos concluir que neste caso o algoritmo pertence a O(V^2).
- Apresentação dos dados. O(1)

A complexidade global da solução será O(min(S, C) * N * M) pois V = 2 * N * M + 2, ou seja O(V) = O(N * M).

Avaliação Experimental dos Resultados:

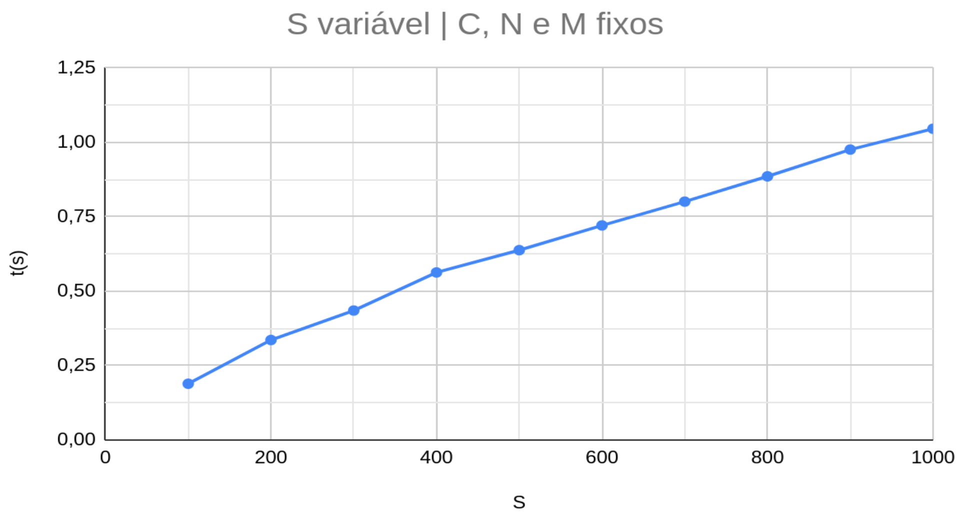
Para avaliar a veracidade dos resultados teóricos, fizemos várias experiências para cada uma das variáveis que apareceram na análise assintótica da complexidade do código.

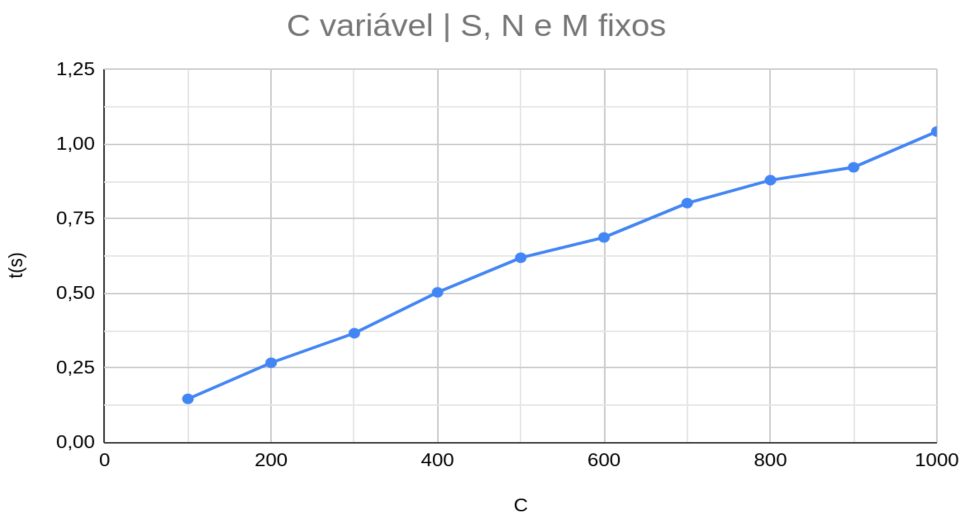
Relatório 2º Projeto ASA 2019/2020

Grupo: al102

Aluno(s): Diogo Barbosa (92451) e Rafael Gonçalves (92544)

Primeiramente variámos S (respetivamente C), fixando C > S (respetivamente S > C) e N * M >> S + C:





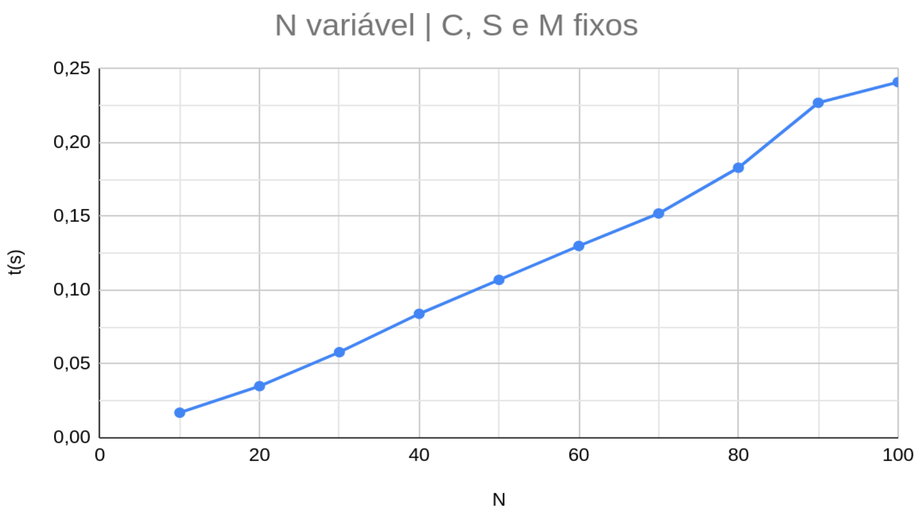
Da análise dos gráficos conclui-se que o tempo depende linearmente de S e C, pelo que dependerá também linearmente de min(S, C).

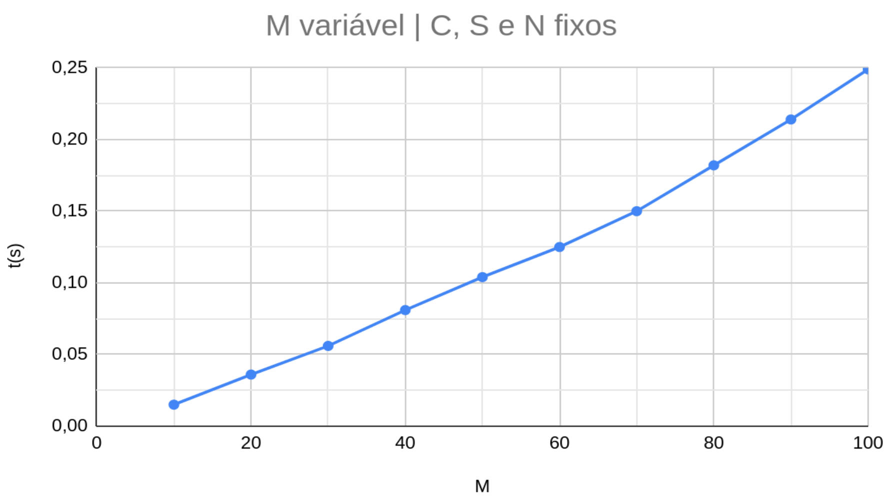
Relatório 2º Projeto ASA 2019/2020

Grupo: al102

Aluno(s): Diogo Barbosa (92451) e Rafael Gonçalves (92544)

Variando agora N e M (sem qualquer relação de grandeza entre eles) com S e C fixos:





Pelo
que o tempo depende linearmente tanto de N como de M.

Uma vez que o tempo de execução apresenta uma relação linear com S, C, N e M através de análise experimental, verificamos a veracidade da complexidade temporal obtida, que corresponde a um produto de variáveis de expoente 1, ou seja, com crescimento linear: O(min(S, C) * N * M).