# 

**Curso:** Ciência de Computação  **Período: 3**º **Turno:** Diurno

**Disciplina:** Estrutura de Dados **Professor:** Fermín Alfredo Tang Montané

**Atividade:** Avaliação AM1 **Data:** 01/04/2021

**Aluno:** João Vítor Fernandes Dias **Matrícula:** 00119110377

**Atividade LE1**

**Esta atividade reúne os exercícios das 3 primeiras semanas. Para simplificar cada questão valerá 1 ponto.**

## Q01

**Considerando os conceitos sobre TAD:**

Desenvolva e implemente um TAD que represente um cilindro. Inclua as funções de inicializações necessárias e as operações que retornem a sua altura e o raio, a sua área e o seu volume. Para cada função, explique brevemente a ideia adotada. Teste o TAD com exemplos. Teste o TAD com uma aplicação.

## Q02

Desenvolva e implemente um TAD que represente um conjunto de inteiros. Para isso, utilize um vetor de inteiros. O TAD deverá conter as seguintes funções:

1. Criar um conjunto vazio. b) Inserir um elemento no conjunto.
2. Remover um elemento no conjunto. d) Testar se um número pertence ao conjunto.
3. Menor valor do conjunto. f) União de dois conjuntos.
4. Testar se os dois conjuntos são iguais. h) Testar se o conjunto é vazio.

Para cada função, explique brevemente a ideia adotada. Teste o TAD com uma aplicação.

## Q03

Especifique um TAD que seja capaz de armazenar uma data composta por dia, mês e ano e implemente o seguinte conjunto de operações para manipular esse tipo Data:

* 1. Uma função que recebe como parâmetro o dia, o mês, e o ano; verifica se a data for válida e converte ela em uma *string* no formato DD/MM/AAAA.
  2. Uma função que recebe como parâmetro uma data de tipo *string* (formato DD/MM/AAAA) e uma estrutura do tipo Data. Armazena os dados da *string* na estrutura.
  3. Uma função que recebe como parâmetro uma estrutura do tipo Data e um número de dias, e retorna a estrutura modificada com soma da data original mais o número de dias recebido.

Teste o TAD com uma aplicação.

## Q04

Os dados relativos aos clientes de uma empresa estão armazenados em um arquivo. Para cada cliente são registrados um código, nome, o endereço, o telefone, a data em que fez sua primeira compra na empresa, a data da última compra e o valor da última compra. Especifique o TAD Clientes para armazenar os dados dos clientes e as operações necessárias para inserir, consultar e excluir esses dados. Implemente uma aplicação que utilize o tipo Clientes.

**Considerando os conceitos de eficiência de algoritmos realize as seguintes tarefas:**

## Q05

Implemente os algoritmos de Soma de Matrizes nxn (quadradas) e Multiplicação de matrizes nxn (quadradas). Os dados das matrizes de entrada podem ser preenchidos de forma aleatória ou usando alguma regra de geradora.

## Q06

Implemente código para medir o desempenho desses algoritmos, considerando:

1. O tempo de execução de cada algoritmo em milissegundos;
2. O número de operações realizadas (somas + multiplicações) por cada algoritmo.

## Q07

Realize experimentos para pelo menos três tamanhos de n. Na escolha dos tamanhos considere que o tamanho deve ser grande o suficiente para permitir a medição de tempo, mas não tanto que possa estourar a memória. Explore isso. Construa uma tabela resumindo os seus resultados.

|  |  |  |
| --- | --- | --- |
| tamanho | tempo soma | tempo Mult |
| 1 | 0 | 0 |
| 2 | 0 | 0 |
| 3 | 0 | 0 |
| 4 | 0 | 0 |
| 5 | 0 | 0 |
| 6 | 0 | 0 |
| 7 | 0 | 0 |
| 8 | 0 | 0 |
| 9 | 0 | 0 |
| 10 | 0 | 0 |
| 20 | 0 | 0 |
| 30 | 0 | 2 |
| 40 | 0 | 0 |
| 50 | 0 | 2 |
| 60 | 0 | 3 |
| 70 | 0 | 4 |
| 80 | 0 | 6 |
| 90 | 0 | 4 |
| 100 | 0 | 6 |
| 110 | 0 | 7 |
| 120 | 0 | 8 |
| 130 | 0 | 10 |
| 140 | 0 | 13 |
| 150 | 0 | 16 |
| 160 | 0 | 19 |
| 170 | 0 | 24 |
| 180 | 0 | 28 |
| 190 | 0 | 32 |
| 200 | 0 | 38 |
| 210 | 0 | 47 |
| 220 | 0 | 51 |
| 230 | 1 | 57 |
| 240 | 0 | 69 |
| 250 | 0 | 86 |
| 260 | 0 | 89 |
| 270 | 0 | 103 |
| 280 | 0 | 110 |
| 290 | 1 | 129 |
| 300 | 0 | 144 |
| 301 | 0 | 144 |
| 302 | 0 | 145 |
| 303 | 0 | 147 |
| 304 | 0 | 139 |
| 305 | 0 | 149 |
| 306 | 0 | 153 |
| 307 | 1 | 153 |
| 308 | 0 | 152 |
| 309 | 0 | 153 |
| 310 | 0 | 156 |
| 311 | 0 | 159 |
| 312 | 1 | 157 |
| 313 | 1 | 161 |
| 314 | 1 | 163 |
| 315 | 0 | 173 |
| 316 | 0 | 166 |
| 317 | 0 | 168 |
| 318 | 0 | 173 |
| 319 | 0 | 176 |
| 320 | 0 | 175 |
| 321 | 0 | 180 |

## Q08

Responda se os seus resultados corroboram as funções O() teóricas de cada algoritmo.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **EXPERIMENTAL** | | | | |
| **Tamanho *n*** | **Soma de Matrizes** | | **Multiplicação de Matrizes** | |
|  | *Tempo (ms)* | *N° Oper.* | *Tempo (ms)* | *N° Oper.* |
| 100 | 0 | 10000 | 5 | 2000000 |
| 300 | 0 | 90000 | 145 | 54000000 |
| 500 | - | - | - | - |
| ... |  |  |  |  |
| 1.000 | - | - | - | - |

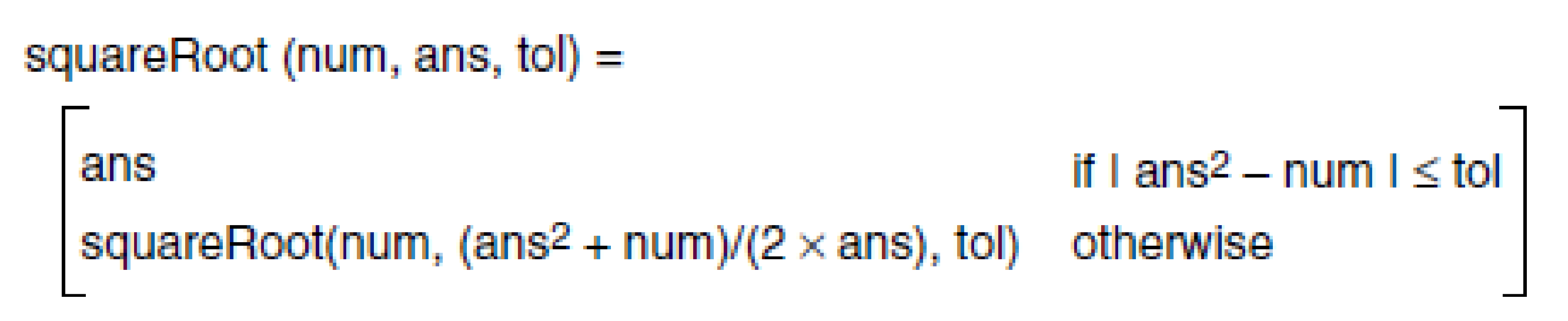
Obs.: As operações feitas na multiplicação de matrizes têm o dobro das operações pois são feitas operações de multiplicação e soma.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Big O** | | | | |
| **Tamanho *n*** | **Soma de Matrizes**  **O(n)=N²** | | **Multiplicação de Matrizes**  **O(N³)** | |
|  | *Tempo (ms)* | *N° Oper.* | *Tempo (ms)* | *N° Oper.* |
| 100 | 10 | 10000 | 1000 | 1000000 |
| 300 | 90 | 90000 | 27000 | 27000000 |
| 500 | 250 | 250000 | 125000 | 125000000 |
| ... |  |  |  |  |
| 1.000 | 1000 | 1000000 | 1000000 | 1000000000 |

## Q09

**Considerando os conceitos sobre recursão:**

Um dos métodos para calcular a raiz quadrada de um número é o método de Newton. A fórmula para o método de newton é mostrada a seguir:



Onde: : é o número cuja raiz quadrada queremos calcular;

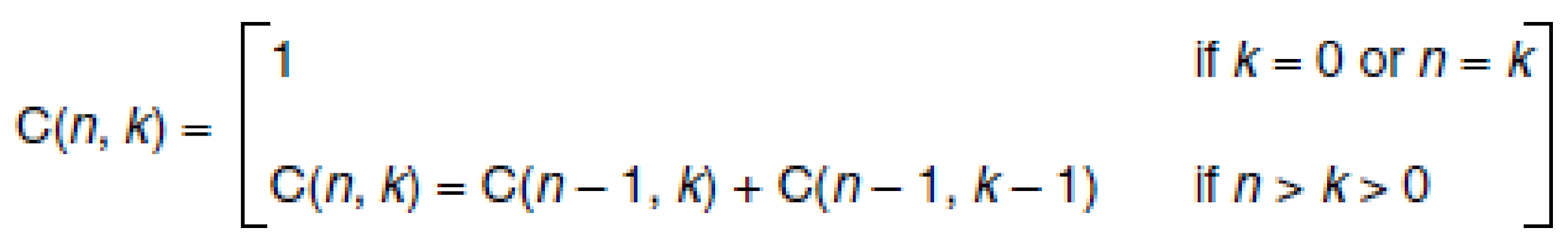
: é uma aproximação inicial da raiz quadrada;

: é a tolerância permitida para a raiz quadrada.

Implemente um algoritmo recursivo que utilize o método de Newton para calcular a raiz quadrada de um número. Verifique o seu algoritmo nos seguintes casos: ;

## Q10

A combinação de objetos tomados de cada vez, pode ser calculada de maneira recursiva usando a seguinte formula de recorrência:



Esta fórmula pode ser usada para calcular o número de combinações possíveis, de objetos. Implemente um algoritmo recursivo para calcular a combinação de objetos tomados de cada vez. Verifique o seu algoritmo, para que corresponde a um total de combinações diferentes.