

ESPECIFICAÇÕES TAD & EXECUÇÃO

EXERCÍCIO 1

TAD **complexo**:

Dados: Duas variáveis **real** e **imag** do tipo double para a parte real e imaginária de um número complexo, respectivamente.

Lista de operações: **cria_nro**, **libera_nro**, **soma**, **sub**, **mult**.

Operações:

cria_nro:

- Entrada: Dois valores do tipo double **real** e **imag** para inicialização do número complexo.
- Pré-condição: Nenhuma.
- Processo: Criar um número complexo com o valor real e imaginário inicializados.
- Saída: Endereço de memória do número complexo criado ou o valor NULL se ocorrer erro na alocação.
- Pós-condição: Nenhuma.

libera_nro:

- Entrada: Endereço do endereço de número complexo.
- Pré-condição: Nenhuma.
- Processo: Libera a memória do número complexo criado e limpa seu endereço.
- Saída: Nenhuma.
- Pós-condição: Número complexo liberado da memória.

soma:

- Entrada: Endereço de dois números complexos a e b.
- Pré-condição: Número complexo ser válido.
- Processo: Calcular a soma entre dois números complexos.
- Saída: NULL se ocorrer algum erro ou a endereço do número complexo com o resultado se der certo.
- Pós-condição: Nenhuma.

sub:

- Entrada: Endereço de dois números complexos a e b.
- Pré-condição: Número complexo ser válido.
- Processo: Calcular a diferença entre o primeiro e o segundo número complexo.
- Saída: NULL se ocorrer algum erro ou a endereço do número complexo com o resultado se der certo.
- Pós-condição: Nenhuma.

ESPECIFICAÇÕES TAD & EXECUÇÃO

mult:

- Entrada: Endereço de dois números complexos a e b.
- Pré-condição: Número complexo ser válido.
- Processo: Calcular o produto entre dois números complexos.
- Saída: NULL se ocorrer algum erro ou a endereço do número complexo com o resultado se der certo.
- Pós-condição: Nenhuma.

get_complex:

- Entrada: Endereço de um número complexo a, endereço de um double real e o endereço de um double imag.
- Pré-condição: Número complexo ser válido.
- Processo: Pegar o valor real e imaginário do número complexo a e colocar em real e imag, respectivamente.
- Saída: 0 se ocorrer algum erro ou 1 se conseguir pegar esses valores.
- Pós-condição: Variáveis real e imag com os valores do número complexo a.

EXECUÇÃO

```
Digite a parte real do numero complexo 1: 1
Digite a parte imaginaria do numero complexo 1: 2

Digite a parte real do numero complexo 2: 2
Digite a parte imaginaria do numero complexo 2: 1

Digite a parte real do numero complexo 3: 2
Digite a parte imaginaria do numero complexo 3: 1

O resultado e: -3.00 + (1.00)i
Process returned 0 (0x0)   execution time : 7.650 s
Press any key to continue.
```

ESPECIFICAÇÕES TAD & EXECUÇÃO

EXERCÍCIO 2

TAD Ponto:

Dados: Três variáveis X, Y e Z do tipo double para as coordenadas

Lista de operações: Cria_pto, Libera_pto, Distancia_pto.

Operações:

Cria_pto:

- Entrada: Três valores do tipo double x, y e z para inicialização do ponto.
- Pré-condição: Nenhuma.
- Processo: Criar um ponto no R3 com os valores x, y e z inicializados.
- Saída: Endereço de memória do ponto criado ou o valor NULL se ocorrer erro na alocação.
- Pós-condição: Nenhuma.

Libera_pto:

- Entrada: Endereço do endereço de um ponto.
- Pré-condição: Nenhuma.
- Processo: Libera a memória do ponto criado e limpa seu endereço.
- Saída: Nenhuma.
- Pós-condição: Ponto liberado da memória.

Distancia_pto:

- Entrada: Endereço de dois pontos p1 e p2.
- Pré-condição: Ponto ser válido.
- Processo: Calcular a distância entre dois pontos do plano cartesiano.
- Saída: -1 se ocorrer algum erro ou a distância entre dois pontos se der certo.
- Pós-condição: Nenhuma.

EXECUÇÃO

```
Digite as coordenadas do ponto 1!
Digite x, y e z: 0 0 0

Digite as coordenadas do ponto 2!
Digite x, y e z: 1 1 1

A distancia entre esses dois pontos e: 1.732
Process returned 0 (0x0)   execution time : 5.481 s
Press any key to continue.
```

ESPECIFICAÇÕES TAD & EXECUÇÃO

TAD esfera:

Dados: Dois endereços do tipo Ponto (p1 e p2).

Lista de operações: cria_esfera, libera_esfera, raio, area, volume.

Operações:

cria_esfera:

- Entrada: Seis valores do tipo double x1, y1, z1(para p1) e x2, y2, z2(para p2), para inicialização da esfera.
- Pré-condição: Nenhuma.
- Processo: Criar uma esfera com um ponto no centro e outro na superfície, com seus devidos valores inicializados.
- Saída: Endereço de memória da esfera criada ou o valor NULL se ocorrer erro na alocação.
- Pós-condição: Nenhuma.

libera_esfera:

- Entrada: Endereço do endereço de uma esfera.
- Pré-condição: Nenhuma.
- Processo: Libera a memória da esfera criada e limpa seu endereço.
- Saída: Nenhuma.
- Pós-condição: Esfera liberada da memória.

raio:

- Entrada: Endereço de uma esfera.
- Pré-condição: Esfera ser válida.
- Processo: Calcular o raio da esfera.
- Saída: -1 se ocorrer algum erro ou o valor do raio.
- Pós-condição: Nenhuma.

area:

- Entrada: Valor do raio de uma esfera.
- Pré-condição: Raio ser positivo.
- Processo: Calcular a área da esfera.
- Saída: -1 se raio for negativo ou o valor da área.
- Pós-condição: Nenhuma.

volume:

- Entrada: Valor do raio de uma esfera.
- Pré-condição: Raio ser positivo.
- Processo: Calcular a área da esfera.
- Saída: -1 se raio for negativo ou o valor do volume.
- Pós-condição: Nenhuma.

ESPECIFICAÇÕES TAD & EXECUÇÃO

EXECUÇÃO

```
Digite (x, y, z) do centro do raio: 0 0 0
Digite (x, y, z) de um ponto da extremidade: 1 1 1

O tamanho do raio e: 1.73
A area da esfera e: 37.70
O volume da esfera e: 21.77
Process returned 0 (0x0)   execution time : 4.536 s
Press any key to continue.
```