

Descrição do Trabalho

Você deve implementar um conjunto de programas que realizam operações com números complexos. Cada número complexo é expresso como um arquivo binário formado por dois doubles, o primeiro para a parte real e o segundo para a parte imaginária.

Relembrando os Números Complexos

Um número complexo é um número $z \in \mathbb{C}$ formado por uma parte real e uma parte imaginária. Há três maneiras de representar um número complexo. A primeira, denotada por representação geométrica, expressa um número complexo como um vetor em um espaço bidimensional, onde a abscissa corresponde a parte real e a ordenada a parte imaginária. A segunda maneira é conhecida como forma algébrica e tem a forma $z = a + bi$, onde o símbolo i expressa que b é o valor imaginário e a a parte real. A terceira forma é chamada de representação trigonométrica ou forma polar. Nessa representação, o número complexo $z = r \cdot e^{i\theta}$ é descrito pelo módulo do vetor que o representa, $r = |z|$ e pelo ângulo θ que faz com o semi-eixo positivo das abscissas, de modo que $r = \sqrt{a^2 + b^2}$ e $\theta = \tan\left(\frac{b}{a}\right)^{-1}$. A Figura 1 apresenta a relação entre as três representações citadas.

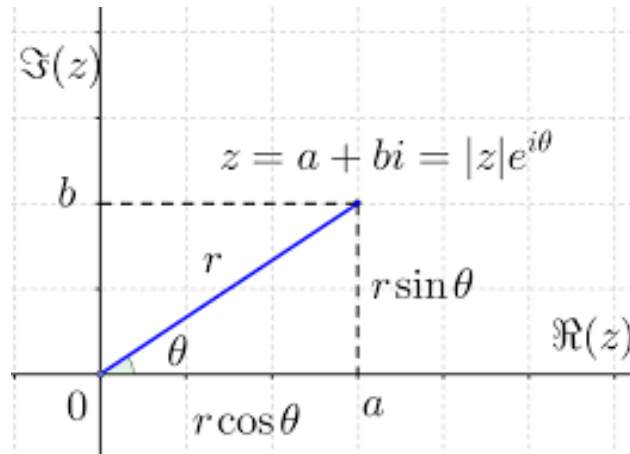


Figura 1. Relação entre as representações de um número complexo.

Formato do arquivo

Neste exercício, um número complexo em arquivo é sempre armazenado na sua representação algébrica em um arquivo binário. Assim, define-se o seguinte formato de arquivo com a estrutura:

- 2 bytes com o conteúdo 'C' e 'P', representando que o arquivo em questão é para armazenar números complexos
- 1 double para armazenar a parte imaginária

- 1 double para armazenar a parte real

A estrutura a ser usada deve ser a seguinte:

```
typedef struct {
    char code[2];
    double real;
    double img;
} Complex;
```

Programas a serem implementados:

1. cpbuild

Este programa recebe, via linha de comando, uma `string` contendo um número complexo e o nome do arquivo que ele deve ser armazenado. O uso desse programa deve seguir o seguinte padrão:

```
cpbuild [OPTION] <complex_number_string> <file_name>

OPTION:
  -a      forma algébrica (default)
  -p      forma polar/trigonométrica
  -v      forma vetorial/geométrica

<complex_number_string>:
  a+bi    para a forma algébrica
  r;theta para a forma polar/trigonométrica
  [a,b]   para a forma vetorial/geométrica

<file_name>:
  nome do arquivo
```

Exemplos de execução:

```
./cpbuild -a 2+3i cp1
./cpbuild -p 5;90 cp2
./cpbuild -v [2,3] cp3
```

A primeira linha de comando cria o arquivo `cp1` com o número complexo $z = 2 + 3i$ usando a forma algébrica.

A segunda linha de comando cria o arquivo `cp2` com o número complexo $z = 5.e^{i\frac{\pi}{2}}$ usando a forma polar. Nota: $\frac{\pi}{2}rads = 90^\circ$.

A terceira linha de comando cria o arquivo `cp3` com o número complexo $z = 2 + 3i$ usando a forma vetorial.

2. cpview

Este programa recebe uma `string` com o nome do arquivo que contém um número complexo e o apresenta no terminal. O uso desse programa deve seguir o seguinte padrão:

```
cpview [OPTION] <file_name>

OPTION:
  -a      forma algébrica (default)
  -p      forma polar/trigonométrica
  -v      forma vetorial/geométrica

<file_name>:
  nome do arquivo
```

Exemplos de execução:

```
$/cpview -a cp1
$ 2+3i
$/cpview -v cp1
$ [2,3]
$/cpview -p cp1
$ 3.60;56.30
```

3. cpsum, cpsub, cpmult e cpdiv

Estes programas realizam a soma, a subtração, a multiplicação e a divisão de dois números complexos e salva o resultado em um arquivo. Ele recebe o nome de três arquivos, sendo os dois primeiros os arquivos que contém os números complexos a serem somados e o último nome representa o nome do arquivo resultante. O uso desse programa deve seguir o seguinte padrão:

```
cpsum <file_name_1> <file_name_2> <file_name_out>
cpsub <file_name_1> <file_name_2> <file_name_out>
cpmult <file_name_1> <file_name_2> <file_name_out>
cpdiv <file_name_1> <file_name_2> <file_name_out>

<file_name_1>:
  nome do arquivo do primeiro número complexo

<file_name_2>:
  nome do arquivo do segundo número complexo

<file_name_out>:
  nome do arquivo para o número complexo resultante
```

Testando o seu trabalho

Seu trabalho será testado por um `script bash` que realiza uma sequência de comandos no terminal chamando os programas que você codificou com os argumentos apropriados. Abaixo segue um exemplo de `script bash` para teste do seu conjunto de programas:

```
#!/usr/bin/bash

./cpbuild -a 3+4i cp1
./cpview -v cp1
./cpbuild -v [3,4] cp2
./cpview -p cp2
./cpbuild -p 5;53.13 cp3
./cpview -v cp3
./cpsum cp1 cp2 cp_sum_1_2
./cpsub cp1 cp3 cp_sub_1_3
./cpmult cp2 cp3 cp_mult_2_3
./cpdiv cp2 cp3 cp_div_2_3
./cpview -a cp_sum_1_2
./cpview -v cp_sub_1_3
./cpview -p cp_mult_2_3
./cpview -a cp_div_2_3
```

Entrega do trabalho

O trabalho é individual e substitui a nota da prova-05 que cobraria o conteúdo de **Arquivos**. O trabalho deve ser entregue em um arquivo zipado (.zip) contendo os códigos de cada programa até o dia 4 de Janeiro de 2021 via o link de submissão no SIGAA.

Dicas e observações

1. Fique a vontade para colocar mensagens de erro que desejar
2. Lembre-se que todo dado é passado por linha de comando. Assim, não há necessidade de leitura do terminal, ou seja, não é necessário o uso do `scanf`
3. Para converter `strings` em double você pode usar a função que implementou nas listas de exercícios ou usar a função `atof` presente na `stdlib.h`
4. O desenvolvimento de todos os programas fica mais fácil se você construir uma biblioteca de números complexos, formada pelos arquivos `complex.h` e `complex.c`. Desse modo, para cada programa a ser implementado, você inclui o seu `header` e chama as funções apropriadas
5. Estou assumindo que você já saiba como lidar com números complexos, no entanto, caso não lembre com detalhes como as operações são feitas, nos links a seguir você pode encontrar uma breve revisão.

Números complexos: definição, operações e exercícios




Os números complexos são números compostos por uma parte real e uma imaginária. Eles representam o conjunto de todos os pares ordenados (x, y) , cujos elementos pertencem ao conjunto dos

 <https://www.todamateria.com.br/numeros-complexos/>



Números complexos: definição, operações, exemplos - Brasil Escola

Os números complexos surgem a partir da necessidade de resolução de equações que possuem raiz de números negativos, o que, até então, não era possível de resolver-se trabalhando com os números reais.

 <https://brasilecola.uol.com.br/matematica/numeros-complexos.htm>

Número complexo

$$a + ib$$

 **Parte real**  **Parte imaginária**