

# MC102 - Algoritmos e Programação de Computadores

MC102

Horários

Plano de  
desenvolvimentoPlano de  
aulasOferecimentos  
anteriores

## Mercado de moedas

---



O Banco Central do Brasil disponibiliza um [Portal de Dados Abertos](#) com indicadores de crédito, de finanças públicas e de atividades econômicas para livre utilização. Nesta tarefa, iremos explorar dados sobre a cotações de moedas que foram publicados recentemente neste portal e pré-processados especialmente para este exercício. As moedas que iremos trabalhar são:

- Dólar Americano (USD)
- Dólar Canadense (CAD)
- Euro (EUR)
- Franco Suíço (CHF)
- Iene (JPY)
- Libra Esterlina (GBP)

A partir de um histórico com cotações em relação ao Real, você deverá calcular o valor mínimo para compra, o valor máximo para venda, a média das cinco cotações mais recentes e a média das cotações de todos os dias fornecidos. A comparação entre estas duas médias indicará uma tendência de alta ou baixa do mercado da moeda sendo analisada.

## Descrição da entrada

---

A primeira linha da entrada conterá a sigla que identifica a moeda. A segunda linha conterá a data da cotação mais recente no formato "AAAA-MM-DD". As linhas seguintes conterão os valores das cotações para compra e venda da moeda com relação ao real. O primeiro par de cotações refere-se ao fechamento na data apresentada, o segundo par ao dia útil anterior a esta data e assim sucessivamente. O histórico termina com um par de zeros. Veja o exemplo:

```
EUR
2019-08-23
4.5526 4.5541
4.4834 4.4856
4.4659 4.4682
4.4825 4.4847
0 0
```

```
4.4673 4.4696
4.4279 4.4290
4.4614 4.4637
4.4644 4.4662
4.4430 4.4441
4.4795 4.4818
0.0000 0.0000
```

## Descrição da saída

---

A saída será composta por seis linhas que serão descritas a seguir. Note que os dados serão precedidos por strings explicativas, escritas propositalmente sem acentos. Todos os valores devem ser formatados com quatro casas decimais.

- **Sigla da moeda:**

```
Moeda: <sigla_da_moeda>
```

- **Data da cotação mais recente:**

```
Data: <data>
```

- **Valor mínimo para compra:**

```
Valor minimo para compra: <valor_min_compra>
```

- **Valor máximo para venda:**

```
Valor maximo para venda: <valor_max_venda>
```

- **Médias das cinco cotações mais recentes:**

```
Medias das cinco cotacoes mais recentes: <media_cinco_compra
media_cinco_venda>
```

- **Médias do historico:**

```
Medias do historico: <media_hist_compra media_hist_venda>
```

Para o exemplo da seção anterior, a saída será:

```
Moeda: EUR
Data: 2019-08-23
Valor minimo para compra: 4.4279
Valor maximo para venda: 4.5541
Medias das cinco cotacoes mais recentes: 4.4903 4.4924
Medias do historico: 4.4728 4.4747
```

## Dicas de Python 3 para esta tarefa:

---

- Como você já sabe, para ler um valor do tipo `float` da entrada você pode utilizar o comando:

```
valor = float(input())
```

- Para ler dois valores do tipo `float` dispostos na mesma linha e separados por espaços em branco, você pode escrever:

```
valor_compra, valor_venda = map(float, input().split())
```

- Para escrever um valor formatado com quatro casas decimais você pode usar o comando `format()`:

```
print(format(valor, ".4f"))
```

Veja outras formas de formatação na página [Formatação em Python](#).

- Você pode desenvolver uma solução eficiente para esta tarefa sem precisar armazenar os dados em uma lista. O **uso de listas** e métodos que operam sobre listas é **permitido**, mas é importante que você saiba resolver o problema sem estes recursos adicionais.

## Testes com o SuSy

---

No SuSy, para cada tarefa, criamos um conjunto de testes com arquivos de entrada `arq<i>.in` e para cada um deles temos uma saída esperada `arq<i>.res`. Consulte os arquivos no SuSy ou no pacote [aux05.zip](#) para ver o conteúdo dos seis testes abertos. Haverá também dois testes fechados.

Leia instruções para fazer os testes em [Testes com o SuSy](#).

## Orientações para submissão

---

Veja [aqui](#) a página de submissão da tarefa. O arquivo a ser submetido deve se chamar `lab05.py`. No link [Arquivos auxiliares](#) há um arquivo [aux05.zip](#) que contém todos os arquivos de testes abertos e seus respectivos resultados compactados.

Utilize o sistema SuSy com o mesmo login e senha que você utiliza para fazer acesso ao sistema da DAC. Se você não estiver inscrito corretamente, envie email para [islene@ic.unicamp.br](mailto:islene@ic.unicamp.br).

O limite máximo será de 15 submissões. Serão considerados os resultados da última submissão.

O peso desta tarefa é 2.

O prazo final para submissão é 15/09/2019.

A nota desta tarefa é proporcional ao número de testes que executaram corretamente, desde que o código esteja coerente com o enunciado. **A submissão de um código que não implementa o algoritmo requisitado, mas que exibe as saídas esperadas dos testes abertos a partir da comparação de trechos da entrada será considerada fraude e acarretará a atribuição de nota zero à média final da disciplina.**

---

A figura que ilustra esta tarefa foi obtida do site <https://buildingsuccessfulonlinebusiness.com>.