



**Fundação CECIERJ - Vice-Presidência de Educação Superior a Distância**

**Curso de Tecnologia em Sistemas de Computação**

**Disciplina Fundamentos de Programação**

**Professores: Dante Corbucci Filho e Luís Felipe Ignácio Cunha**

**APX2 1º semestre de 2021**

---

### **IMPORTANTE**

**MUITA ATENÇÃO:** Nesta APX há, na próxima página, o **TERMO de CONDUTA** que todo aluno deve estar ciente e assinar devidamente o documento. Você pode redigi-lo e assinar manuscrito ou assinar digitalmente, e deve fazer upload deste termo juntamente com o arquivo ZIP das suas respostas. Caso o **TERMO de CONDUTA** não seja devidamente entregue e assinado, a APX1 não será corrigida e a nota da APX2 será zero.

- As respostas (programas) deverão ser entregues pela plataforma em um arquivo ZIP contendo todos os arquivos de código fonte (extensão “.py”) necessários para que os programas sejam testados. Respostas entregues fora do formato especificado, por exemplo, em arquivos com extensão “.pdf”, “.doc” ou outras, não serão corrigidas.
  - Serão aceitos apenas soluções escritas na linguagem Python 3. Programas com erro de interpretação não serão corrigidos. Evite problemas utilizando tanto a versão da linguagem de programação (Python 3.X) quanto a IDE (PyCharm) indicadas na Aula 1.
  - Quando o enunciado de uma questão inclui especificação de formato de entrada e saída, tal especificação deve ser seguida à risca pelo programa entregue. Atender ao enunciado faz parte da avaliação e da composição da nota final.
  - Os exemplos fornecidos nos enunciados das questões correspondem a casos específicos apontados para fins de ilustração e não correspondem ao universo completo de entradas possíveis especificado no enunciado. Os programas entregues devem ser elaborados considerando qualquer caso que siga a especificação e não apenas os exemplos dados. Essa é a prática adotada tanto na elaboração das listas exercícios desta disciplina quanto no mercado de trabalho.
  - Faça uso de boas práticas de programação, em especial, na escolha de identificadores de variáveis, subprogramas e comentários no código.
  - As respostas deverão ser entregues pela atividade específica na Plataforma antes da data final de entrega estabelecida. Não serão aceitas entregas tardias ou substituição de respostas após término do prazo.
  - As APXs são um mecanismo de avaliação individual. As soluções podem ser buscadas por grupos de alunos, mas a redação final de cada prova tem que ser individual. Respostas plagiadas não serão corrigidas.
-



## TERMO DE CONDUTA

Declaro assumir o compromisso de confidencialidade e de sigilo escrito, fotográfico e verbal sobre as questões do exame ou avaliação pessoal que me serão apresentadas, durante o curso desta disciplina. Comprometo-me a não revelar, reproduzir, utilizar ou dar conhecimento, em hipótese alguma, a terceiros, e a não utilizar tais informações para gerar benefício próprio ou de terceiros. Reitero minha ciência de que não poderei fazer cópia manuscrita, registro fotográfico, filmar ou mesmo gravar os enunciados que me são apresentados. Declaro, ainda, estar ciente de que o não cumprimento de tais normas caracteriza infração ética, podendo acarretar punição de acordo com as regras da minha universidade.

Ciente,

Assinatura do Aluno

### Questão 1 – (Valor 3.0 pontos)

Faça um programa em Python contendo subprograma(s), que leia da entrada padrão o nome de um arquivo texto contendo em cada linha um resultado da Olimpíada de Tóquio:

nome do país#tipo da medalha#modalidade

(1) Leia apenas uma vez cada linha do arquivo e vá produzindo três conjuntos (tipo set) de medalhas, chamados ouro, prata e bronze, onde o nome do país é a informação em cada conjunto.

(2) Mostre o conjunto dos países que obtiveram medalha(s) de ouro. Em seguida mostre a listagem dos países com medalha(s) de prata. Para finalizar, mostre todos os países com medalha(s) de bronzes. Todo(s) escrito(s) um país por linha.

(3) Escreva todos os países que tiveram pelo menos uma medalha. Todo(s) escrito(s) um país por linha.

(4) Escreva o(s) nome(s) do(s) país(es) que teve(tiveram) pelo menos uma medalha de cada tipo (ouro, prata e bronze). Todo(s) escrito(s) um país por linha.

Restrição: Em geral, está proibido manter o conteúdo de todo de um arquivo em memória principal, isto é, em listas, strings ou similares.

Conteúdo do Arquivo medalhas:

```
brasil#prata#judô masculino -70k
japão#ouro#judô masculino -70k
eua#bronze#judô masculino -70k
frança#bronze#judô masculino -70k
brasil#ouro#judô feminino -70k
brasil#bronze#judô feminino -70k
```

### Teste:

Entradas :	Saídas Correspondentes :
medalhas	<pre>Países com Ouro(s) :     brasil     japão  Países com Prata(s) :     brasil  Países com Bronze(s) :     brasil     eua     França  Países com pelo menos uma medalha:     brasil     eua     França     japão  País(es) com pelo menos uma Medalha de cada Tipo:</pre>

	brasil
--	--------

## Questão 2 – (Valor 3.0 pontos)

Faça um programa que processe arquivos texto contendo em cada linha os dados de infectados pelo Covid-19. Cada linha contém:

cfp#cepa#data#cidade#estado

Seu programa deve pedir inicialmente o nome do arquivo, o nome de infectados e qual cepa deseja-se informações. Produza e mostre o dicionário (dict) onde a chave é o nome do estado e o valor é outro dicionário (dict) de cidades que possuam infectados daquele estado cuja a chave é a cidade e o valor é o número de infectados em cada mês, ordenado pelos anos e meses em cada cidade. Ao final, mostre o total de infectados por aquela cepa escolhida.

Conteúdo do Arquivo alfa:

```
497290403#indiana#15/4/2021#Niterói#RJ
765839293#inglesa#15/4/2021#Maricá#RJ
497295403#indiana#22/3/2021#Niterói#RJ
497297403#inglesa#19/3/2021#São Gonçalo#RJ
777777777#inglesa#1/1/2021#Araraquara#SP
555555555#indiana#1/2/2021#Niterói#RJ
333333333#indiana#11/12/2020#Niterói#RJ
111111111#indiana#11/2/2021#Niterói#RJ
```

## Teste:

Entradas:	Saídas Correspondentes:
alfa inglesa	Tipo da Cepa: inglesa RJ Maricá Ano: 2021 Mês: 4 Infectados: 1 São Gonçalo Ano: 2021 Mês: 3 Infectados: 1 SP Araraquara Ano: 2021 Mês: 1 Infectados: 1  Total Geral: 3
alfa indiana	Tipo da Cepa: indiana RJ Niterói Ano: 2020 Mês: 12 Infectados: 1 Ano: 2021 Mês: 2 Infectados: 2 Ano: 2021 Mês: 3 Infectados: 1 Ano: 2021 Mês: 4 Infectados: 1  Total Geral: 5

### Questão 3 – (Valor 4.0 pontos)

Faça um programa, que a partir de um arquivo .bin consistindo da codificação de um catálogo de séries disponíveis nos serviços de diferentes plataformas de streaming, retorne resultados de consultas em relação ao ano de produção das séries produzidas, série a ser buscada e conteúdo das plataforma de streaming que buscamos.

A entrada é dada por: (1) um arquivo binário, (2) um tipo de consulta e (3) uma consulta específica de acordo com o tipo apresentado em (2).

(1) Arquivo binário:

O arquivo binário de nome "Series.bin" é composto por uma sequência de registros onde a cada linha há informações sobre ano de produção da série, título da série, plataforma de streaming onde a série se passa. A separação entre as informações é feita com o símbolo #. Notem que uma série pode ser transmitida por mais do que uma plataforma de streaming, dessa forma, deve haver um registro por linha.

O ano ocupa 4 bytes, o nome da série ocupa 30 bytes e a plataforma de streaming 60 bytes. Todos os dados são do tipo string.

(2) Tipo de consulta:

Na entrada há três tipos de consulta possíveis:

Ano: Isso faz retornar todas as séries do ano xxxx consultado, separadas pelo serviço de streaming, da seguinte forma:

Deve ter na saída o nome de cada serviço de streaming e suas correspondentes séries produzidas no ano determinado. A exibição dos serviços de streaming devem estar em ordem alfabética, assim como a ordem de listagem das séries também devem estar em ordem alfabética.

Caso o ano consultado xxxx não possua na lista de catálogo, então o retorno deve ser: No catálogo apresentado não há séries produzidas no ano de xxxx.

Serie: Isso faz retornar todos os serviço de streaming que transmitem a série xxxx, da seguinte forma:

Deve ter na saída a listagem de todas as plataformas de streaming cuja série esteja em cartaz. O retorno deve conter o ano de produção da série desejada e a listagem dos serviços de streaming. Os serviços de streaming devem aparecer necessariamente em ordem alfabética. Além disso, você deve considerar na listagem as vírgulas e 'e' entre os dois últimos:

Exemplo:

Elementary possui produção de 2012 e está em cartaz em: Globoplay e Prime Video.

Caso a série consultada xxxx não possua na lista de catálogo, então o retorno deve ser: xxxx não está listado no catálogo.

Plataforma: Isso faz retornar todas séries transmitidas pela plataforma xxxx, separadas por ano, da seguinte forma:

Os anos devem aparecer em ordem crescente, e para cada ano, as séries devem estar em ordem alfabética.

Caso a plataforma consultada xxxx não esteja na lista de catálogo, então o retorno deve ser: xxxx não está listado no catálogo.

Caso o tipo xxxx escrito na entrada não seja de nenhum dos três tipos apresentados, então a saída deve ser: xxxx não é um tipo possível a ser consultado.

Vocês devem apresentar nas soluções os arquivos: .txt a ser codificado, .py que codifica para o .bin e o .py que faz a programação desejada.

As entradas, saídas correspondentes e o arquivo decodificado do .bin devem ser conforme os testes abaixo:

**Teste:**

Entradas:	Saídas Correspondentes:
Series.bin Ano 2017	Globoplay The Good Doctor Netflix Dark La Casa de Papel
Series.bin Ano 2020	No catálogo apresentado não há séries produzidas no ano de 2020.
Series.bin Serie Elementary	Elementary possui produção de 2012 e está em cartaz em: Globoplay e Prime Video.
Series.bin Serie Monk	Monk não está listado no catálogo.
Series.bin Plataforma Netflix	2016 Stranger Things 2017 Dark La Casa de Papel 2019 O Segredo do Templo 2021 Lupin
Series.bin Plataforma HBO GO	HBO GO não está listado no catálogo.
Series.bin Ano 2017	Ano não é um tipo possível a ser consultado.
<b>Arquivo: (Series.bin)</b>	
2017#Dark#Netflix 2017#The Good Doctor#Globoplay 2017#La Casa de Papel#Netflix 2016#Stranger Things#Netflix 2019#O Segredo do Templo#Netflix 2021#Lupin#Netflix 2012#Elementary#Globoplay 2012#Elementary#Prime Video 2005#Everybody Hates Chris#Prime Video 2005#Everybody Hates Chris#Globoplay	

**Obs.: O arquivo Series.bin aparece no exemplo dessa forma para melhor compreensão da questão.**

**Boa Avaliação!**