

Questão 1  
Incompleto  
Vale 10,00  
ponto(s).  
 Marcar questão

## PyDiner Dash

Dêivis está querendo começar sua carreira como empreendedor. Ele pretende abrir seu primeiro restaurante na pandemia! Refletindo nas últimas vezes que Dêivis pediu sua ajuda, dessa vez ele decidiu pedir sua ajuda antes que tudo desse errado! Ajude Dêivis a construir, gerir e analisar as estatísticas da ocupação de um restaurante!

### Objetivo

O objetivo deste projeto é que o aluno aprenda a desenvolver um programa na linguagem Python utilizando o conceito de variáveis, captura e escrita de informações na tela, condicionais, iteração, funções, listas e boas práticas de programação.

### Entrada e Comandos

Os dados a serem fornecidos tem o formato abaixo:

```
--CONFIGURACAO
a m c
a m c
.
.
.
--ATENDIMENTO
comando
texto (existirá caso o comando exija)
.
.
onde:
```

- **--CONFIGURACAO** marca o início da seção de configuração do espaço do restaurante. A configuração ocorrerá até que comence a seção de atendimentos do restaurante.
- **a** indica a área que existirá no restaurante. **a** é uma string,  $1 \leq \text{quantidade}(a) \leq 10$
- **m** indica quantas mesas existirão naquela área. **m** é um inteiro  $1 \leq m \leq 100$
- **c** indica quantas cadeiras cada mesa daquela área terá. **c** é um inteiro  $1 \leq c \leq 100$
- **--ATENDIMENTO** marca o início da seção de atendimentos do restaurante. Os atendimentos ocorrerão até que seja encontrado o input de comando **-1**.
- **comando** indica qual comando de atendimento será executado naquele momento. **comando** é um inteiro tal que  $\text{comando} \in \{1, 2, 3, 4, -1\}$ .
- **texto** indica um texto que alguns comandos irão precisar para funcionarem corretamente.
- **-1** marca o fim do programa.

A seguir há uma descrição de cada comando e a entrada adicional de texto, caso este necessite de uma.

1. Colocar um grupo de pessoas em uma mesa (comando = 1)

O comando 1 consiste em alocar uma mesa na área indicada restaurante para um grupo de pessoas. Para isso, é necessária uma linha abaixo do comando com o texto no seguinte formato:

Quero uma mesa para X pessoas na area Y

onde X indica quantas pessoas tem no grupo e Y indica qual a área o grupo quer ocupar.

### \*\*\*Particularidades do Comando\*\*\*

Assuma que não serão requeridas mesas em áreas que não existem.

Além da implementação simples acima, o comando possui algumas particularidades tanto na entrada quanto no comportamento dos grupos que estão ocupando as mesas, que serão explicadas a seguir:

- Organização otimizada

Como dito acima, você deve sempre alocar a menor mesa possível que ainda acomode um grupo de pessoas, para otimizar o uso dos lugares disponíveis no restaurante.

- Permanência dos grupos

Como um restaurante é rotativo, ou seja, as pessoas tem que sair do restaurante para que outras entrem, cada grupo de pessoas ficará por um número finito de comandos, determinado pela equação

$$\text{tempo\_de\_permanencia} = \text{quantidade(comandos)} = (2 * X) + 2$$

onde X é a quantidade de pessoas no grupo, e comandos são os comandos executados pelo usuário no input (incluindo este que está sendo explicado). Cada comando é uma unidade de tempo de permanência, ou seja, um casal que consiga uma mesa no restaurante permanecerá nele por até 6 comandos, ou seja, 6 unidades de tempo. É claro que, quando o restaurante fecha, não haverá mais comandos (mas os clientes poderão terminar suas refeições tranquilamente). O próprio comando de alocar uma mesa no restaurante para um grupo de pessoas também contará para o tempo de permanência.

## 2. Consulta de mesas (comando = 2)

Imprime como saída uma lista com cada área, em ordem alfabética, com o respectivo número de mesas ocupadas de um total de mesas.

## 3. Consulta de lotação (comando = 3)

Imprime como saída uma lista com cada área, em ordem alfabética, com a respectiva lotação atual e capacidade total.

## 4. Adição ou remoção de mesas (comando = 4)

O comando 4 consiste em adicionar ou remover mesas com quantidade específica de cadeiras em uma área. Para isso, é necessária uma linha abaixo do comando com o texto no seguinte formato:

Quero OP mais Z mesas com X cadeiras cada na area Y

onde OP ∈ {adicionar, remover}, Z é a quantidade de mesas a serem adicionadas, X é a quantidade de cadeiras de cada mesa, e Y é a área em que as mesas serão adicionadas/removidas.

### \*\*\*Particularidades do comando\*\*\*

Assuma que nunca serão adicionadas mesas em áreas não existentes, nem que serão removidas mesas que estão ocupadas ou que não existem!!!!!!

## 5. Fechar restaurante e consulta da configuração final do restaurante (comando = -1)

Neste comando, é necessário imprimir a configuração final do restaurante, após a adição e remoção de mesas (caso tenham ocorrido), separadas por área (ordenadas em ordem alfabética), com a configuração de mesas e cadeiras (ordenadas em ordem crescente de cadeiras), e a quantidade de pessoas que entraram no restaurante.

## Saída

Como dito acima, na seção de atendimentos, serão executados comandos para o restaurante. A seguir há uma descrição da saída desejada baseada em cada comando inserido.

### 1. Colocar um grupo de pessoas em uma mesa (comando = 1)

- Caso haja uma mesa com lugares o suficiente para acomodar a quantidade de pessoas, deve imprimir:

Um grupo de **X** pessoas ocupou uma mesa de **L** lugares na área **Y**.

onde **L** é a menor quantidade de lugares que uma mesa possui no espaço **Y**, que também pode acomodar um grupo de **X** pessoas;

- Caso não tenha uma mesa com lugares o suficiente para acomodar a quantidade de pessoas (seja pela lotação ou por não ter nenhuma mesa grande o suficiente para o grupo de pessoas), deve imprimir:

Nao foi possível levar o grupo de clientes para uma mesa.

### 2. Consulta de mesas (comando = 2)

Imprime como saída uma lista com cada área, **em ordem alfabética**, com o respectivo número de mesas ocupadas de um total de mesas, no seguinte formato:

A1: (**M1** de **T1** mesas)  
A2: (**M2** de **T2** mesas)  
A3: (**M3** de **T3** mesas)  
.  
.  
An: (**Mn** de **Tn** mesas)

onde **A** é a área existente, **M** é a quantidade de mesas ocupadas na área e **T** é a quantidade de mesas totais da área.

### 3. Consulta de lotação (comando = 3)

Imprime como saída uma lista com cada área, **em ordem alfabética**, com a respectiva lotação atual e capacidade total, no seguinte formato:

A1: (**C1** de **T1** pessoas)  
A2: (**C2** de **T2** pessoas)  
A3: (**C3** de **T3** pessoas)  
.  
.  
An: (**Cn** de **Tn** pessoas)

onde **A** é a área existente, **C** é a quantidade de cadeiras ocupadas na área e **T** é a quantidade de cadeiras totais da área.

### 4. Adição ou remoção de mesas (comando = 4)

Imprima na tela a mensagem de confirmação:

**Z** mesas de **X** cadeiras **OP** com sucesso na área **Y**.

onde **OP** ∈ {*adicionadas*, *removidas*}, **Z** é a quantidade de mesas a serem adicionadas, **X** é a quantidade de cadeiras de cada mesa, e **Y** é a área em que as mesas foram adicionadas/removidas.

### 5. Fechar restaurante e consulta da configuração final do restaurante (comando = -1)

É necessário imprimir a configuração final do restaurante, após a adição e remoção de mesas (caso tenham ocorrido), separadas por área (**ordenadas em ordem alfabética**), com a configuração de mesas e cadeiras (**ordenadas em ordem crescente de cadeiras**), e a quantidade de pessoas que entraram no restaurante, com o seguinte formato:

```

Restaurante fechado.
Balanco final de mesas:
A1:
M1 mesas de C1 cadeiras.
M2 mesas de C2 cadeiras.
M3 mesas de C3 cadeiras.
A2:
M4 mesas de C4 cadeiras.
M5 mesas de C5 cadeiras.
.
.
.

Um total de T pessoas visitaram o restaurante hoje.
Bom descanso!

```

onde A são as áreas, M são quantas mesas de uma quantidade específica de cadeiras a área tem, C são quantas cadeiras cada mesa tem e T são quantas pessoas visitaram o restaurante.

### \*\*\*OBSERVAÇÕES ÚTEIS\*\*\*

Para o tratamento das strings, pode-se utilizar o método `.split()`, e para a ordenação pedida nos comandos 2, 3 e -1, recomenda-se utilizar o método `.sort()` ou utilizar a função `sorted()`.

For example:

Input	Result
--CONFIGURACAO VARANDA 1 20 VARANDA 5 4 SALAO 4 2 SALAO 2 4 SALAO 1 8 --ATENDIMENTO -1	Restaurante fechado. Balanco final de mesas: SALAO: 4 mesas de 2 cadeiras. 2 mesas de 4 cadeiras. 1 mesas de 8 cadeiras. VARANDA: 5 mesas de 4 cadeiras. 1 mesas de 20 cadeiras. Um total de 0 pessoas visitaram o restaurante hoje. Bom descanso!
--CONFIGURACAO AREA_Z 2 2 AREA_Z 1 4 AREA_Y 4 2 AREA_Y 2 4 AREA_Y 1 8 --ATENDIMENTO 1 Quero uma mesa para 3 pessoas na area AREA_Y -1	Um grupo de 3 pessoas ocupou uma mesa de 4 lugares na area AREA_Y. Restaurante fechado. Balanco final de mesas: AREA_Y: 4 mesas de 2 cadeiras. 2 mesas de 4 cadeiras. 1 mesas de 8 cadeiras. AREA_Z: 2 mesas de 2 cadeiras. 1 mesas de 4 cadeiras. Um total de 3 pessoas visitaram o restaurante hoje. Bom descanso!
--CONFIGURACAO VARANDA 2 2 VARANDA 1 4 SALAO 4 2 EVENTOS 2 4 EVENTOS 1 8 --ATENDIMENTO 1 Quero uma mesa para 3 pessoas na area EVENTOS 1 Quero uma mesa para 3 pessoas na area SALAO 	Um grupo de 3 pessoas ocupou uma mesa de 4 lugares na area EVENTOS. Nao foi possivel levar o grupo de clientes para uma mesa. Restaurante fechado. Balanco final de mesas: EVENTOS: 2 mesas de 4 cadeiras. 1 mesas de 8 cadeiras. SALAO: 4 mesas de 2 cadeiras. VARANDA: 2 mesas de 2 cadeiras. 1 mesas de 4 cadeiras. Um total de 3 pessoas visitaram o restaurante hoje. Bom descanso!
--CONFIGURACAO AREA1 4 2 AREA1 2 4 AREA1 1 8 AREA2 2 2	AREA0: (0 de 40 pessoas) AREA1: (0 de 24 pessoas) AREA2: (0 de 8 pessoas) AREA4: (0 de 24 pessoas) AREA5: (0 de 8 pessoas)

AREA2 1 4 AREAS 1 8 AREA4 4 6 AREA0 10 4 -- ATENDIMENTO 3 -1	Restaurante fechado. Balanco final de mesas: AREA0: 10 mesas de 4 cadeiras. AREA1: 4 mesas de 2 cadeiras. 2 mesas de 4 cadeiras. 1 mesas de 8 cadeiras. AREA2: 2 mesas de 2 cadeiras. 1 mesas de 4 cadeiras. AREA4: 4 mesas de 6 cadeiras. AREAS: 1 mesas de 8 cadeiras. Um total de 0 pessoas visitaram o restaurante hoje. Bom descanso!
--CONFIGURACAO SALAO 4 2 SALAO 2 4 SALAO 1 8 VARANDA 2 2 VARANDA 1 4 --ATENDIMENTO 4 Quero adicionar mais 3 mesas com 4 cadeiras cada na area SALAO -1	3 mesas de 4 cadeiras adicionadas com sucesso na area SALAO. Restaurante fechado. Balanco final de mesas: SALAO: 4 mesas de 2 cadeiras. 5 mesas de 4 cadeiras. 1 mesas de 8 cadeiras. VARANDA: 2 mesas de 2 cadeiras. 1 mesas de 4 cadeiras. Um total de 0 pessoas visitaram o restaurante hoje. Bom descanso!
--CONFIGURACAO SALAO 4 2 SALAO 2 4 SALAO 1 8 VARANDA 2 2 VARANDA 1 4 --ATENDIMENTO 4 Quero remover mais 2 mesas com 2 cadeiras cada na area SALAO -1	2 mesas de 2 cadeiras removidas com sucesso na area SALAO. Restaurante fechado. Balanco final de mesas: SALAO: 2 mesas de 2 cadeiras. 2 mesas de 4 cadeiras. 1 mesas de 8 cadeiras. VARANDA: 2 mesas de 2 cadeiras. 1 mesas de 4 cadeiras. Um total de 0 pessoas visitaram o restaurante hoje. Bom descanso!
--CONFIGURACAO FUMANTES 4 2 FUMANTES 2 4 FUMANTES 1 8 NAO-FUMANTES 2 2 NAO-FUMANTES 1 4 --ATENDIMENTO 1 Quero uma mesa para 2 pessoas na area FUMANTES 2 3 1 Quero uma mesa para 2 pessoas na area FUMANTES 2 3 -1	Um grupo de 2 pessoas ocupou uma mesa de 2 lugares na area FUMANTES. FUMANTES: (1 de 7 mesas) NAO-FUMANTES: (0 de 3 mesas) FUMANTES: (2 de 24 pessoas) NAO-FUMANTES: (0 de 8 pessoas) Um grupo de 2 pessoas ocupou uma mesa de 2 lugares na area FUMANTES. FUMANTES: (2 de 7 mesas) NAO-FUMANTES: (0 de 3 mesas) FUMANTES: (4 de 24 pessoas) NAO-FUMANTES: (0 de 8 pessoas) Restaurante fechado. Balanco final de mesas: FUMANTES: 4 mesas de 2 cadeiras. 2 mesas de 4 cadeiras. 1 mesas de 8 cadeiras. NAO-FUMANTES: 2 mesas de 2 cadeiras. 1 mesas de 4 cadeiras. Um total de 4 pessoas visitaram o restaurante hoje. Bom descanso!
--CONFIGURACAO SALAO 4 4 --ATENDIMENTO 1 Quero uma mesa para 3 pessoas na area SALAO 1 Quero uma mesa para 4 pessoas na area SALAO 2 2 2 2 2 2 2 2	Um grupo de 3 pessoas ocupou uma mesa de 4 lugares na area SALAO. Um grupo de 4 pessoas ocupou uma mesa de 4 lugares na area SALAO. SALAO: (2 de 4 mesas) SALAO: (1 de 4 mesas) SALAO: (1 de 4 mesas) SALAO: (0 de 4 mesas) SALAO: (0 de 4 mesas) SALAO: (0 de 4 mesas) Restaurante fechado.

2 2 2 2 -1	Balanco final de mesas: SALAO: 4 mesas de 4 cadeiras. Um total de 7 pessoas visitaram o restaurante hoje. Bom descanso!
--CONFIGURACAO SALAO 1 2 SALAO 2 3 SALAO 3 4 SALAO 4 5 SALAO 1 10 VARANDA 1 9 VARANDA 9 1 EVENTOS 10 10 NAO-FUMANTES 1 1 --ATENDIMENTO 1 Quero uma mesa para 2 pessoas na area SALAO 1 Quero uma mesa para 2 pessoas na area SALAO 1 Quero uma mesa para 3 pessoas na area SALAO 1 Quero uma mesa para 3 pessoas na area SALAO 1 Quero uma mesa para 13 pessoas na area SALAO 2 3 4 Quero adicionar mais 1 mesas com 15 cadeiras cada na area SALAO 1 Quero uma mesa para 13 pessoas na area SALAO 2 3 -1	Um grupo de 2 pessoas ocupou uma mesa de 2 lugares na area SALAO. Um grupo de 2 pessoas ocupou uma mesa de 3 lugares na area SALAO. Um grupo de 3 pessoas ocupou uma mesa de 3 lugares na area SALAO. Um grupo de 3 pessoas ocupou uma mesa de 4 lugares na area SALAO. Nao foi possivel levar 5 EVENTOS: (0 de 10 mesas) NAO-FUMANTES: (0 de 1 mesas) SALAO: (4 de 11 mesas) VARANDA: (0 de 10 mesas) EVENTOS: (0 de 100 pessoas) NAO-FUMANTES: (0 de 1 pessoas) SALAO: (8 de 50 pessoas) VARANDA: (0 de 18 pessoas) 1 mesas de 15 cadeiras adicionadas com sucesso na area SALAO. Um grupo de 13 pessoas ocupou uma mesa de 15 lugares na area SALAO. EVENTOS: (0 de 10 mesas) NAO-FUMANTES: (0 de 1 mesas) SALAO: (3 de 12 mesas) VARANDA: (0 de 10 mesas) EVENTOS: (0 de 100 pessoas) NAO-FUMANTES: (0 de 1 pessoas) SALAO: (16 de 65 pessoas) VARANDA: (0 de 18 pessoas) Restaurante fechado. Balanco final de mesas: EVENTOS: 10 mesas de 10 cadeiras. NAO-FUMANTES: 1 mesas de 1 cadeiras. SALAO: 1 mesas de 2 cadeiras. 2 mesas de 3 cadeiras. 3 mesas de 4 cadeiras. 4 mesas de 5 cadeiras. 1 mesas de 10 cadeiras. 1 mesas de 15 cadeiras. VARANDA: 9 mesas de 1 cadeiras. 1 mesas de 9 cadeiras. Um total de 23 pessoas visitaram o restaurante hoje. Bom descanso!

**Answer:** (penalty regime: 0, 0, 10, 20, ... %)

1	
---	--

**Verificar**

**Finalizar tentativa ...**

Seguir para...



[Obter o aplicativo para dispositivos móveis](#)

