

Treinamento Maratona de Programação

Extensão – 3o Encontro

Prof. Dr. Alexandre dos Santos Mignon (alexandre.mignon@mackenzie.br)

Prof. Ms. Fabio Lubacheski (fabio.lubacheski@mackenzie.br)

Prof. Dr. Ivan Carlos Alcântara de Oliveira (ivan.oliveira@mackenzie.br)

Prof. Dr. Leandro Carlos Fernandes (leandro.fernandes@mackenzie.br)



Objetivos



Nesta aula temos como meta:

- Apresentar de forma sucinta vetor e matriz.
- Resolver dois problemas Beecrowd um com vetor e outro com Matriz
- Mostrar como tratar o “término da entrada com fim-de-arquivo (EOF)”.
- Destacar algumas submissões de soluções dos problemas da semana anterior para sua reflexão.
- Esclarecimento de dúvidas em geral e considerações finais.



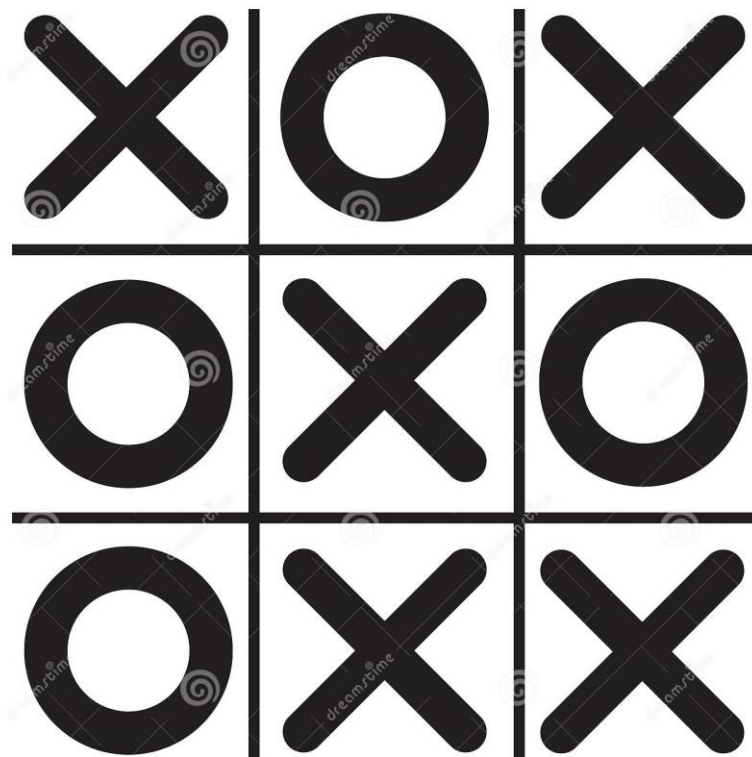
Calendário dos Encontros do Treinamento Maratona de Programação

Encontro(E vento)	Data	Assunto Sugeridos/Previstos	Atividade
01	08/03	Apresentação + Ferramentas	Problemas
02	15/03	Mensagens Beecrowd – Soluções Problemas	Problemas
03	22/03	Vetores e Matrizes	Problemas
04	12/04	Pilha, Fila e outras Estruturas de Dados	Problemas
05	19/04	Algoritmos de Ordenação	Problemas
06	26/04	Algoritmos Gulosos	Problemas
07	03/05	Grafos – Árvore Geradora de Custo Mínimo	Problemas
08	10/05	Recursividade e Backtracking	Problemas
09	17/05	Programação Dinâmica	Problemas
10	24/05	Síntese dos Resultados	Fechamento



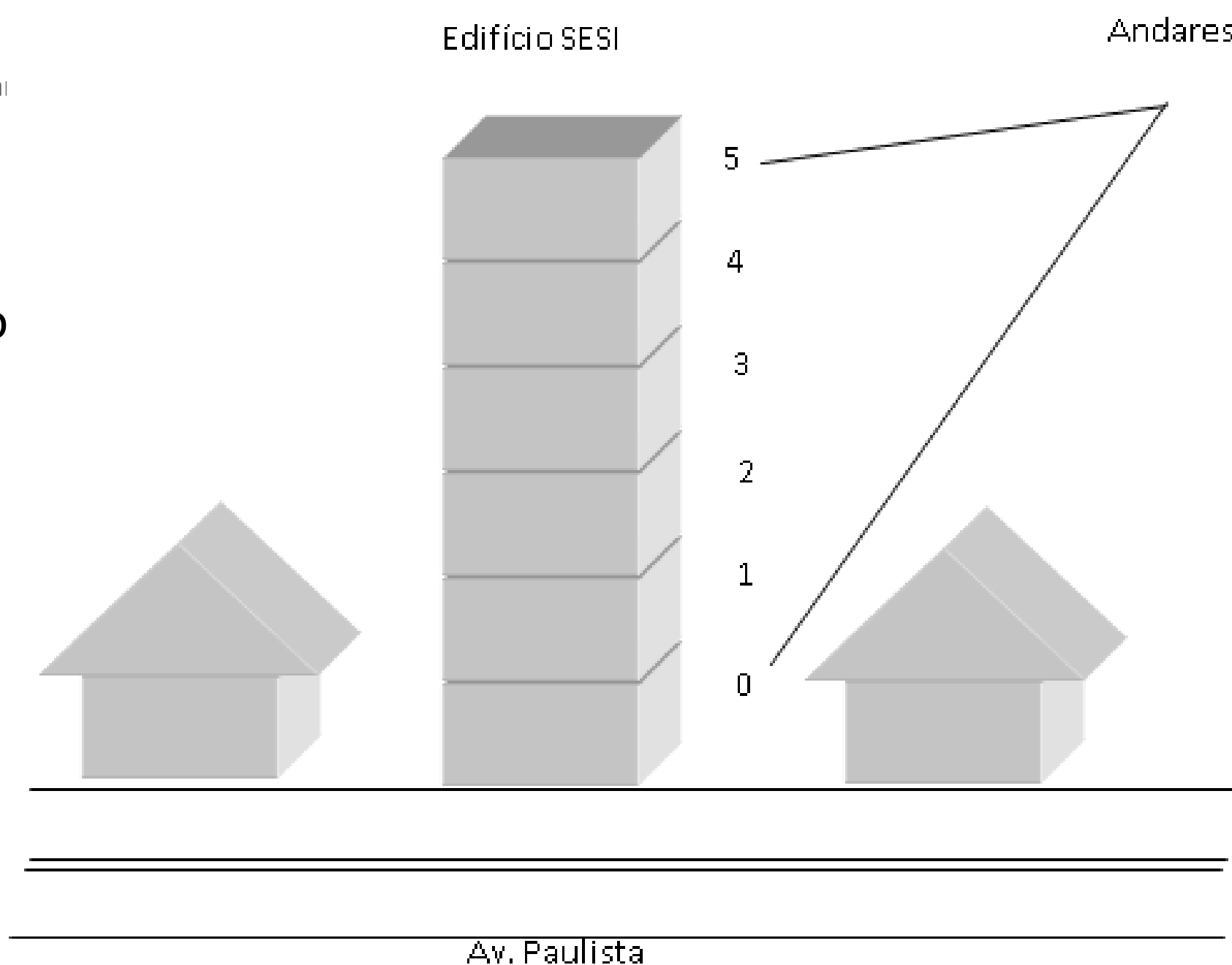
Horário dos Encontros: 15h às 16h.

Vetor e Matriz



Vetor

Para obter o número de pessoas que trabalham/residem no andar 4 basta contar a quantidade de pessoas deste andar.

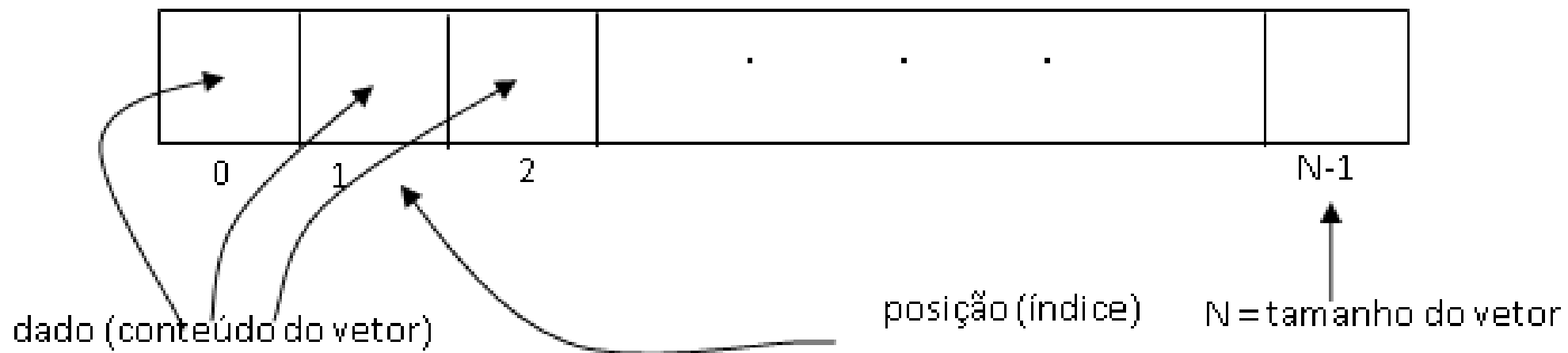


Aspectos Teóricos

- Variáveis compostas homogêneas unidimensionais (vetor) é um conjunto de variáveis onde o conteúdo de cada variável é obtido fornecendo a posição dentro do conjunto.
- Como uma analogia a um vetor, pode-se imaginar um edifício com um número finito de andares.
- Para obter alguma informação sobre um andar específico é necessário informar o número do andar e obter a informação desejada.
- Um vetor é uma estrutura capaz de armazenar informações de um mesmo tipo de dado (no exemplo, a quantidade de pessoas é um valor inteiro).
- Para encontrar um dado que está armazenado neste tipo de estrutura basta informar a posição (índice) onde se encontra o dado.

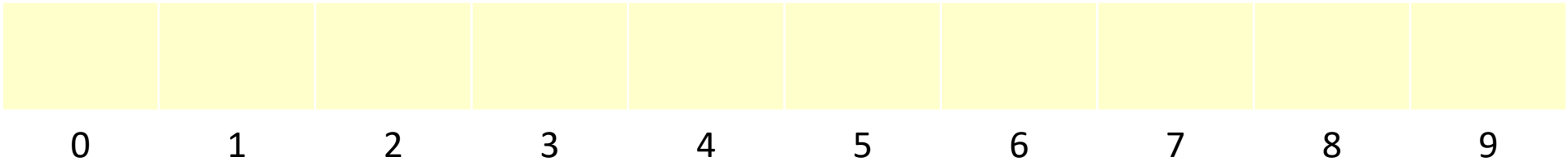


Ilustração



Declaração

Constante inteiro MAX_ANDARES = 10;
inteiro pessoasAndar [MAX_ANDARES];



Exemplo de Uso

Para i de 0 até MAX_ANDARES-1
****pessoasAndar[i] <- i+1;****

1	2	3	4	5	6	7	8	9	10
0	1	2	3	4	5	6	7	8	9

Exemplo Beecrowd

beecrowd | 2310



Voleibol

Por Leonardo Fernandes, IFSC 🇧🇷 Brazil

Timelimit: 1

Um treinador de voleibol gostaria de manter estatísticas sobre sua equipe. A cada jogo, seu auxiliar anota quantas tentativas de saques, bloqueios e ataques cada um de seus jogadores fez, bem como quantos desses saques, bloqueios e ataques tiveram sucesso (resultaram em pontos). Seu programa deve mostrar qual o percentual de saques, bloqueios e ataques do time todo tiveram sucesso.



Entrada

A entrada é dada pelo número de jogadores **N** ($1 \leq N \leq 100$), seguido pelo nome de cada um dos jogadores. Abaixo do nome de cada jogador, seguem duas linhas com três inteiros cada. Na primeira linha **S**, **B** e **A** ($0 \leq S, B, A \leq 10000$) representam a quantidade de tentativas de saques, bloqueios e ataques e na segunda linha, **S1**, **B1** e **A1** ($0 \leq S1 \leq S$; $0 \leq B1 \leq B$; $0 \leq A1 \leq A$) com o número de saques, bloqueios e ataques deste jogador que tiveram sucesso.

Saída

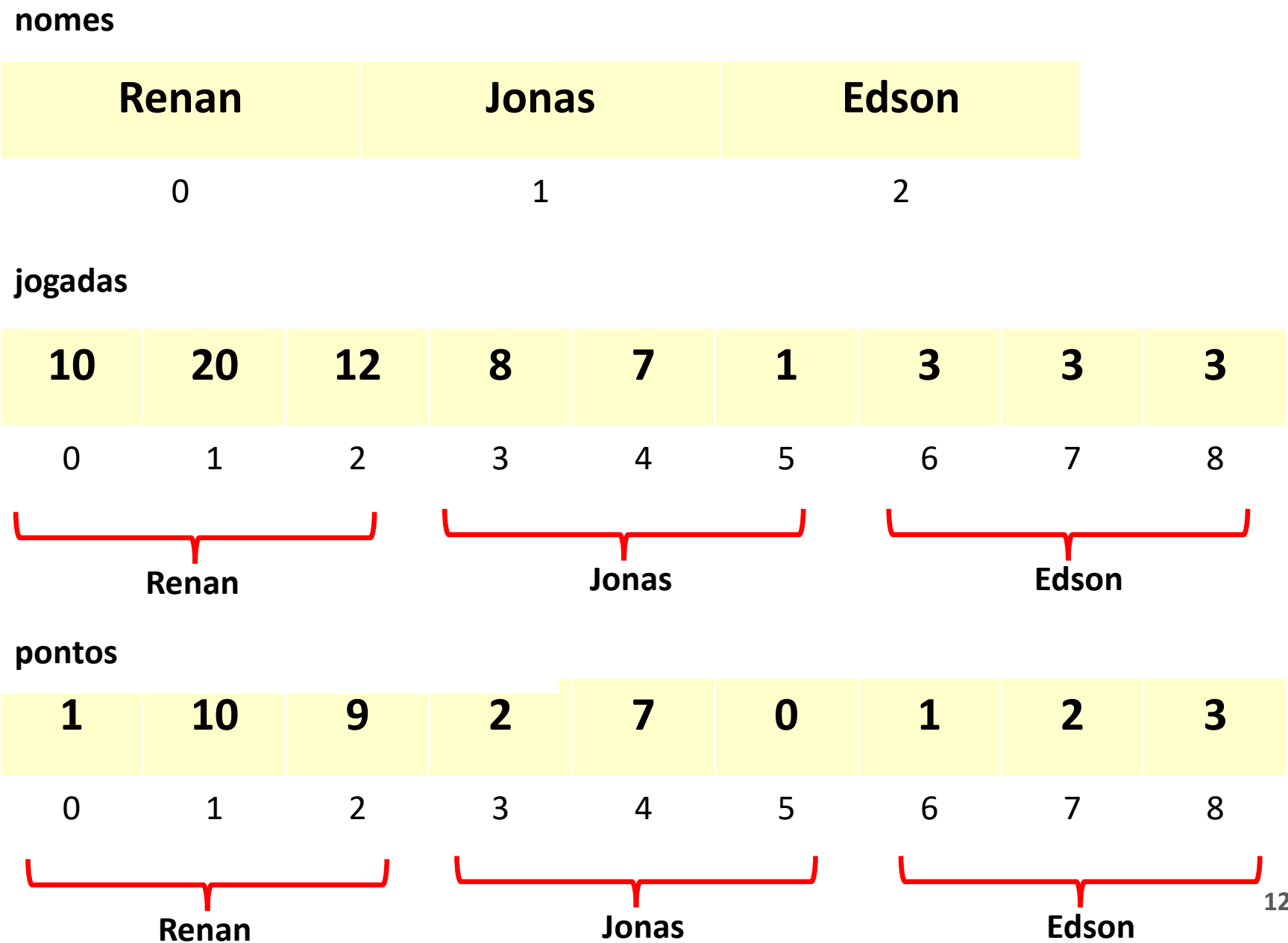
A saída deve conter o percentual total de saques, bloqueios e ataques do time todo que resultaram em pontos, conforme mostrado no exemplo.

Exemplo de Entrada	Exemplo de Saída
3 Renan 10 20 12 1 10 9 Jonas 8 7 1 2 7 0 Edson 3 3 3 1 2 3	Pontos de Saque: 19.05 %. Pontos de Bloqueio: 63.33 %. Pontos de Ataque: 75.00 %.



Resolução

```
3
Renan
10 20 12
1 10 9
Jonas
8 7 1
2 7 0
Edson
3 3 3
1 2 3
```



Resolução Python

```
3
Renan
10 20 12
1 10 9
Jonas
8 7 1
2 7 0
Edson
3 3 3
1 2 3
```



Lê a quantidade de jogadores

```
N = int(input())
```

Cria vetores de nomes, jogadas e pontos

```
nomes = [""]*N
```

```
jogadas = [0]*3*N
```

```
pontos = [0]*3*N
```

Lê nomes, jogadas e pontos

```
for i in range(0,N):
```

```
    nomes[i] = input()
```

Leitura do vetor de jogadas

```
jogada = input().split()
```

```
k = 0
```

```
for j in range(i*3, i*3+3):
```

```
    jogadas[j] = int(jogada[k])
```

```
    k +=1
```

Leitura do vetor de pontos

```
ponto = input().split()
```

```
k = 0
```

```
for j in range(i*3, i*3+3):
```

```
    pontos[j] = int(ponto[k])
```

```
    k +=1
```

Resolução Python

```
3
Renan
10 20 12
1 10 9
Jonas
8 7 1
2 7 0
Edson
3 3 3
1 2 3
```

soma total de jogadas de saque, bloqueio, ataque

```
jogadasSaque = 0
```

```
jogadasBloqueio = 0
```

```
jogadasAtaque = 0
```

```
for i in range(N):
```

```
    jogadasSaque += jogadas[i*3]
```

```
    jogadasBloqueio += jogadas[(i*3)+1]
```

```
    jogadasAtaque += jogadas[(i*3)+2]
```

soma total de pontos de saque, bloqueio, ataque

```
pontosSaque = 0
```

```
pontosBloqueio = 0
```

```
pontosAtaque = 0
```

```
for i in range(N):
```

```
    pontosSaque += pontos[i*3]
```

```
    pontosBloqueio += pontos[(i*3)+1]
```

```
    pontosAtaque += pontos[(i*3)+2]
```



Resolução Python

Exemplo de Saída

```
Pontos de Saque: 19.05 %.  
Pontos de Bloqueio: 63.33 %.  
Pontos de Ataque: 75.00 %.
```

Escreve mensagens de saída

```
print(f"Pontos de Saque: {(pontosSaque/jogadasSaque)*100:.2f} %.")  
print(f"Pontos de Bloqueio: {(pontosBloqueio/jogadasBloqueio)*100:.2f} %.")  
print(f"Pontos de Ataque: {(pontosAtaque/jogadasAtaque)*100:.2f} %.")
```



Resolução Python

Exemplo de Saída

```
Pontos de Saque: 19.05 %.  
Pontos de Bloqueio: 63.33 %.  
Pontos de Ataque: 75.00 %.
```



CÓDIGO FONTE

```
1  #-*- coding: utf-8 -*-  
2  
3  ...  
4  Escreva a sua solução aqui  
5  Code your solution here  
6  Escriba su solución aquí  
7  ...  
8  # Lê a quantidade de jogadores  
9  N = int (input())  
10 # Cria vetores de nomes, jogadas e pontos  
11 nomes = [""]*N  
12 jogadas = [0]*3*N  
13 pontos = [0]*3*N  
14 # Lê nomes, jogadas e pontos  
15 for i in range(0,N):  
16     nomes[i] = input()  
17     # Leitura do vetor de jogadas  
18     jogada = input().split()  
19     k = 0  
20     for j in range(i*3, i*3+3):  
21         jogadas[j] = int(jogada[k])  
22         k +=1  
23     # Leitura do vetor de pontos  
24     ponto = input().split()  
25     k = 0  
26     for j in range(i*3, i*3+3):  
27         pontos[j] = int(ponto[k])  
28         k +=1  
29  
30 # soma total de jogadas de saque, bloqueio, ataque  
31 jogadasSaque = 0  
32 jogadasBloqueio = 0  
33 jogadasAtaque = 0  
34 for i in range(N):  
35     jogadasSaque += jogadas[i*3]  
36     jogadasBloqueio += jogadas[(i*3)+1]  
37     jogadasAtaque += jogadas[(i*3)+2]  
38  
39 # soma total de pontos de saque, bloqueio, ataque  
40 pontosSaque = 0  
41 pontosBloqueio = 0  
42 pontosAtaque = 0  
43 for i in range(N):  
44     pontosSaque += pontos[i*3]  
45     pontosBloqueio += pontos[(i*3)+1]  
46     pontosAtaque += pontos[(i*3)+2]  
47  
48 # Escreve mensagens de saída  
49 print(f"Pontos de Saque: {(pontosSaque/jogadasSaque)*100:.2f} %.")  
50 print(f"Pontos de Bloqueio: {(pontosBloqueio/jogadasBloqueio)*100:.2f} %.")  
51 print(f"Pontos de Ataque: {(pontosAtaque/jogadasAtaque)*100:.2f} %.")
```


Resolução Python

SUBMISSÃO # 32440267

PROBLEMA:	2310 - Voleibol
RESPOSTA:	Accepted
LINGUAGEM:	Python 3.8 (Python 3.8.2) [+1s]
TEMPO:	0.019s
TAMANHO:	1,37 KB
MEMÓRIA:	-
SUBMISSÃO:	21/03/2023 12:01:57

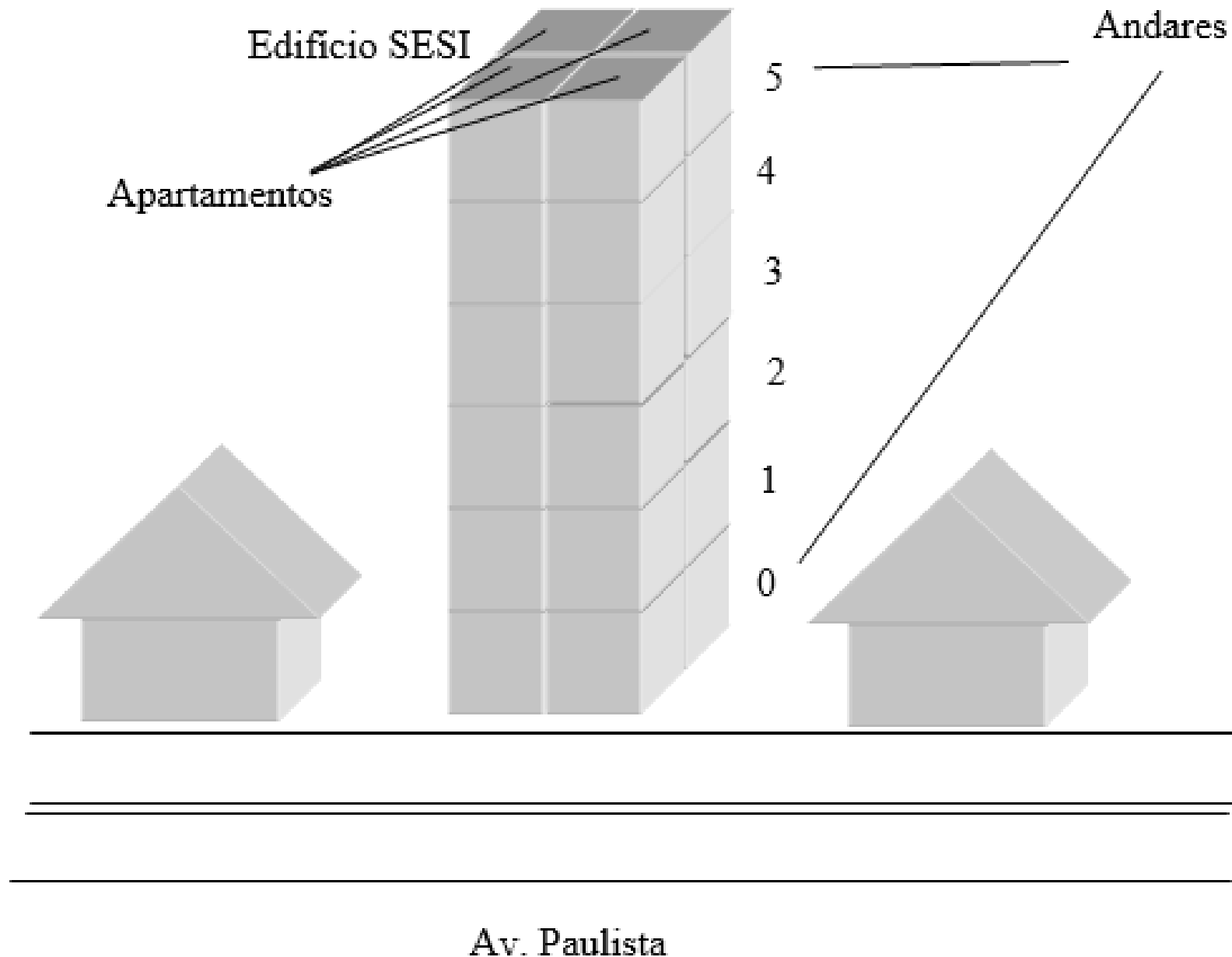
Exemplo de Saída

```
Pontos de Saque: 19.05 %.  
Pontos de Bloqueio: 63.33 %.  
Pontos de Ataque: 75.00 %.
```



Matriz

Para obter alguma informação sobre um apartamento específico basta informar o número do andar e o número do apartamento dentro do andar obtendo, dessa forma, a informação que se deseja.



Aspectos Teóricos

Variáveis compostas homogêneas multidimensionais (matriz) é um conjunto de variáveis onde o conteúdo de cada variável é obtido fornecendo a posição do elemento dentro do conjunto.

A posição é dada por mais de uma coordenada (x, y, z, ...).

Essa estrutura necessita de mais de um índice para obter a informação.

Por exemplo, para encontrar o número de pessoas que habitam/trabalham no apartamento 32 do prédio da figura acima é necessário informar o número do andar (3º) e o número do apartamento (2º).

Como há somente a necessidade de dois índices essa estrutura é chamada bidimensional.



Ilustração

a_{12} = Conteúdo da linha 1 coluna 2

Quantidade de Linhas →

Quantidade de Colunas ↓

	0	1	2	...	M-1
0	a_{00}	a_{01}	a_{02}		$a_{0(M-1)}$
1	a_{10}	a_{11}	a_{12}		$a_{1(M-1)}$
2	a_{20}	a_{21}	a_{22}		$a_{2(M-1)}$
...					
N-1	$a_{(N-1)0}$	$a_{(N-1)1}$	$a_{(N-1)2}$		$a_{(N-1)(M-1)}$

Declaração

- Declaração:

inteiro edificioSESI [6][4];

- O número de elementos (variáveis) desse conjunto é: $6 * 4 = 24$ (apartamentos).

	0	1	2	3
0				
1				
2				
3				
4				
5				



Exemplo de Uso

para i de 0 até 5

para j de 0 até 3

leia(edificioSESI[i][j])

Apartamento 21 do edifício Sesi.

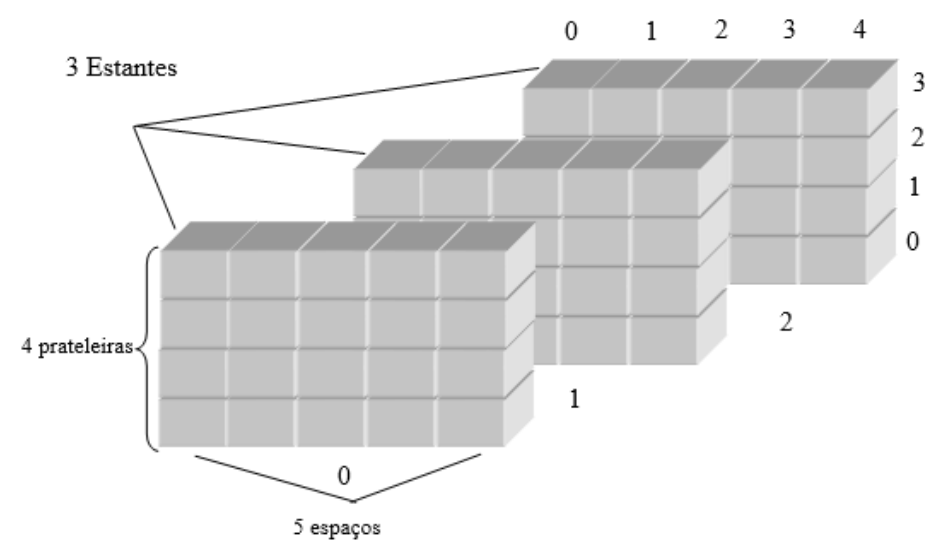
edificioSESI[2][1]

Observa-se que no apartamento
21 há 3 pessoas residindo/trabalhando.

	0	1	2	3
0	3	0	6	5
1	4	4	5	4
2	3	3	5	2
3	1	2	4	1
4	2	3	3	3
5	3	4	2	4

Declaração

inteiro deposito [3][4][5];



- O número de elementos (variáveis) desse conjunto é: $3 * 4 * 5 = 60$ (códigos de peças).

A atribuição: `deposito [1][2][3] <- 10`

Indica que a peça de código 10 está localizada na estante de número 1, prateleira 2, espaço 3 .

Exemplo Beecrowd

beecrowd | 2542



Iu-Di-Oh!

Por Ricardo Oliveira, UFPR  Brazil**Timelimit: 1**

Iu-di-oh! é um jogo de cartas que virou uma verdadeira febre entre os jovens! Todo jogador de *Iu-di-oh!* tem seu próprio baralho, contendo várias cartas do jogo. Cada carta contém **N** atributos (como força, velocidade, inteligência, etc.). Os atributos são numerados de 1 a **N** e são dados por inteiros positivos.

Uma partida de *Iu-di-oh!* é sempre jogada por dois jogadores. Ao iniciar a partida, cada jogador escolhe exatamente uma carta de seu baralho. Após as escolhas, um atributo é sorteado. Vence o jogador cujo atributo sorteado em sua carta escolhida é maior que na carta escolhida pelo adversário. Caso os atributos sejam iguais, a partida empata.

Marcos e Leonardo estão na grande final do campeonato brasileiro de *Iu-di-oh!*, cujo prêmio é um Dainavision (que é quase um Plaisteition 2!). Dados os baralhos de ambos, a carta escolhida por cada um e o atributo sorteado, determine o vencedor!



Entrada

A entrada contém vários casos de teste. A primeira linha de cada caso contém um inteiro **N** ($1 \leq N \leq 100$), o número de atributos de cada carta. A segunda linha contém dois inteiros **M** e **L** ($1 \leq M, L \leq 100$), o número de cartas no baralho de Marcos e de Leonardo, respectivamente.

As próximas **M** linhas descrevem o baralho de Marcos. As cartas são numeradas de 1 a **M**, e a *i*-ésima linha descreve a *i*-ésima carta. Cada linha contém **N** inteiros $a_{i,1}, a_{i,2}, \dots, a_{i,N}$ ($1 \leq a_{i,j} \leq 10^9$). O inteiro $a_{i,j}$ indica o atributo *j* da carta *i*. As próximas **L** linhas descrevem o baralho de Leonardo. As cartas são numeradas de 1 a **L** e são descritas de maneira análoga.

A próxima linha contém dois inteiros **C_M** e **C_L** ($1 \leq C_M \leq M, 1 \leq C_L \leq L$), as cartas escolhidas por Marcos e Leonardo, respectivamente. Por fim, a última linha contém um inteiro **A** ($1 \leq A \leq N$) indicando o atributo sorteado.

A entrada termina com fim-de-arquivo (EOF).

Saída

Para cada caso de teste, imprima uma linha contendo “Marcos” se Marcos é o vencedor, “Leonardo” se Leonardo é o vencedor, ou “Empate” caso contrário (sem aspas).

Exemplo de Entrada	Exemplo de Saída
3 2 2 3 8 1 6 7 9 1 2 3 8 4 1 1 2 2	Marcos



Discussão

A entrada termina com fim-de-arquivo. (EOF)

<https://www.beecrowd.com.br/judge/en/questions/view/2534/1333?origem=1>

Python:

GFAGGIONE REPLIED 5 YEARS AGO

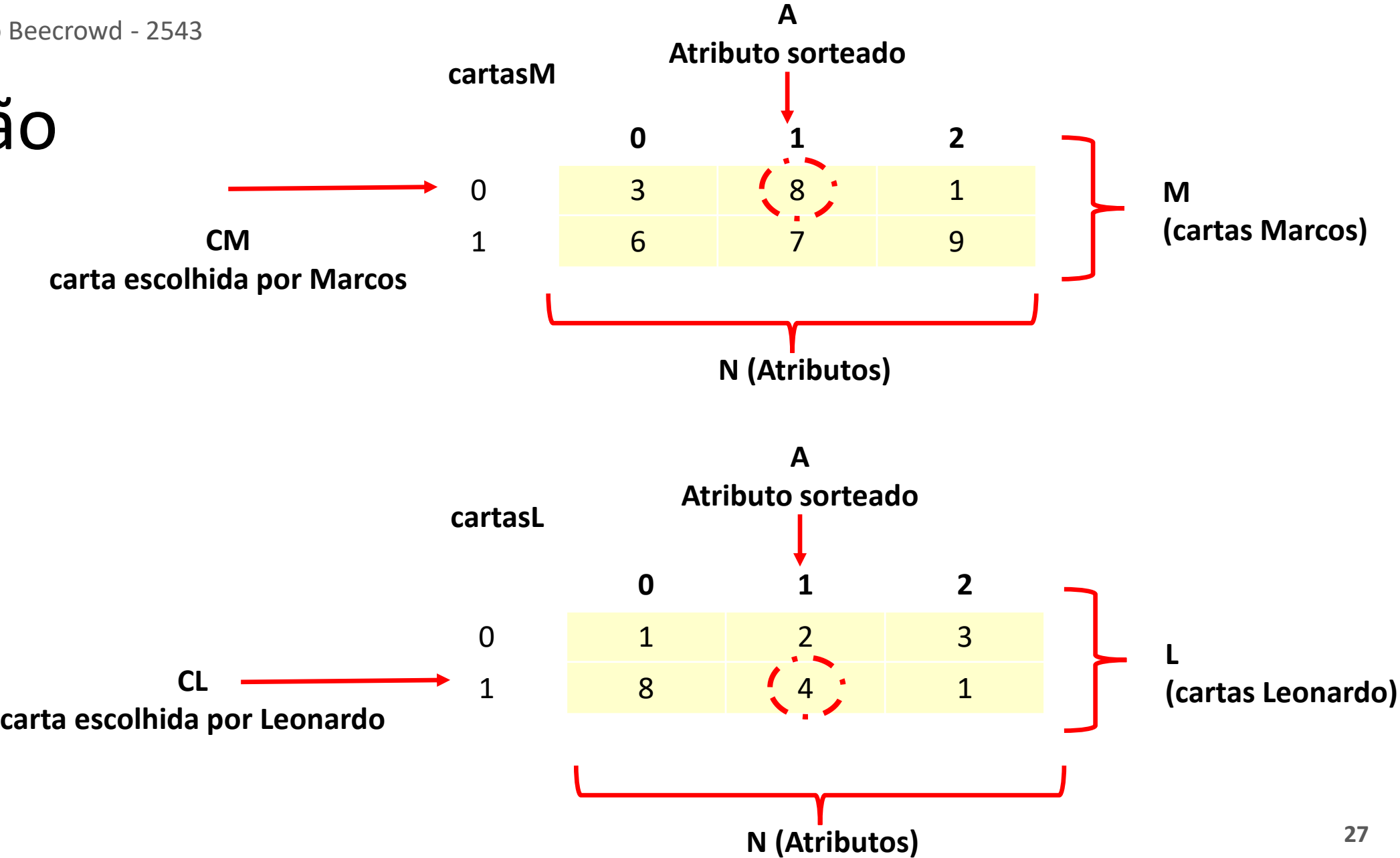
se vc estiver fazendo em python, vc pode fazer assim:

```
while True:
    try:
        #seu código entra aqui
    except EOFError:
        break
```

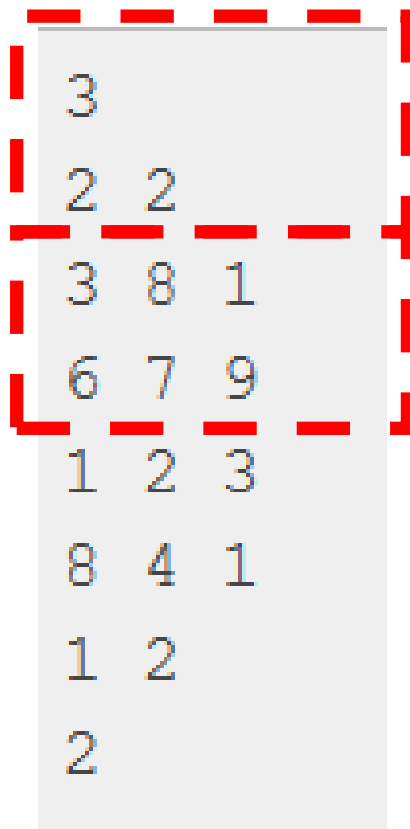


Resolução

3		
2	2	
3	8	1
6	7	9
1	2	3
8	4	1
1	2	
2		



Resolução Python



3		
2	2	
3	8	1
6	7	9
1	2	3
8	4	1
1	2	
2		

while True:

try:

N = int(input()) # Número de atributos

M, L = input().split() # cartas de Marcos e Leonardo

M = int(M)

L = int(L)

cria matriz de cartas de Marcos

cartasM = []

for i in range(M):

cartasM = cartasM + [[0]*N]

leitura das cartas de Marcos

for i in range(M): # número de linhas

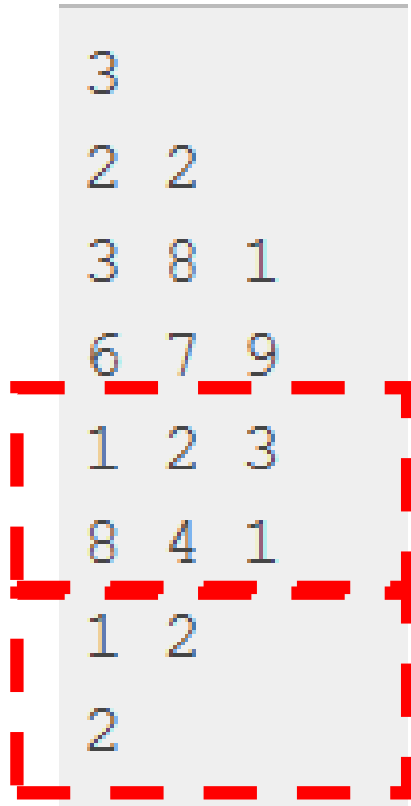
lin = list(input().split())

for j in range(N): # colunas

cartasM[i][j] = int(lin[j])



Resolução Python



3		
2	2	
3	8	1
6	7	9
1	2	3
8	4	1
1	2	
2		

cria matriz de cartas de Leonardo

```
cartasL = []  
for i in range(L):  
    cartasL = cartasL + [[0]*N]
```

leitura das cartas de Leonardo

```
for i in range(L): # número de linhas  
    lin = list(input().split())  
    for j in range(N): # número de colunas  
        cartasL[i][j] = int(lin[j])
```

carta escolhida por Marcos e Leonardo

```
CM, CL = input().split()  
CM = int(CM)  
CL = int(CL)  
A = int(input()) # atributo sorteado
```

Resolução Python

Exemplo de Saída

Marcos

```
# obtém a carta/atributo de Marcos e Leonardo
```

```
cartaM = cartasM[CM-1][A-1]
```

```
cartaL = cartasL[CL-1][A-1]
```

```
if cartaM > cartaL:
```

```
    print("Marcos")
```

```
elif cartaM < cartaL:
```

```
    print("Leonardo")
```

```
else:
```

```
    print("Empate")
```

```
except EOFError:
```

```
    break
```



Resolução

Exemplo de Saída

Marcos

```

1 while True:
2     try:
3         N = int(input()) # Número de atributos
4         M, L = input().split() # cartas de Marcos e Leonardo
5         M = int(M)
6         L = int(L)
7
8         # cria matriz de cartas de Marcos
9         cartasM = []
10        for i in range(M):
11            cartasM = cartasM + [[0]*N]
12
13        # leitura das cartas de Marcos
14        for i in range(M): # número de linhas
15            lin = list(input().split())
16            for j in range(N): # colunas
17                cartasM[i][j] = int(lin[j])
18
19        # cria matriz de cartas de Leonardo
20        cartasL = []
21        for i in range(L):
22            cartasL = cartasL + [[0]*N]
23

```

CÓDIGO FONTE



Resolução

Exemplo de Saída

Marcos

```
24 # leitura das cartas de Marcos
25 for i in range(L): # número de linhas
26     lin = list(input().split())
27     for j in range(N): # número de colunas
28         cartasL[i][j] = int(lin[j])
29 # carta escolhida por Marcos e Leonardo
30 CM, CL = input().split()
31 CM = int(CM)
32 CL = int(CL)
33 A = int(input()) # atributo sorteado
34
35 # obtém a carta/atributo de Marcos e Leonardo
36 cartaM = cartasM[CM-1][A-1]
37 cartaL = cartasL[CL-1][A-1]
38
39 if cartaM > cartaL:
40     print("Marcos")
41 elif cartaM < cartaL:
42     print("Leonardo")
43 else:
44     print("Empate")
45 except EOFError:
46     break
```



Resolução

Exemplo de Saída

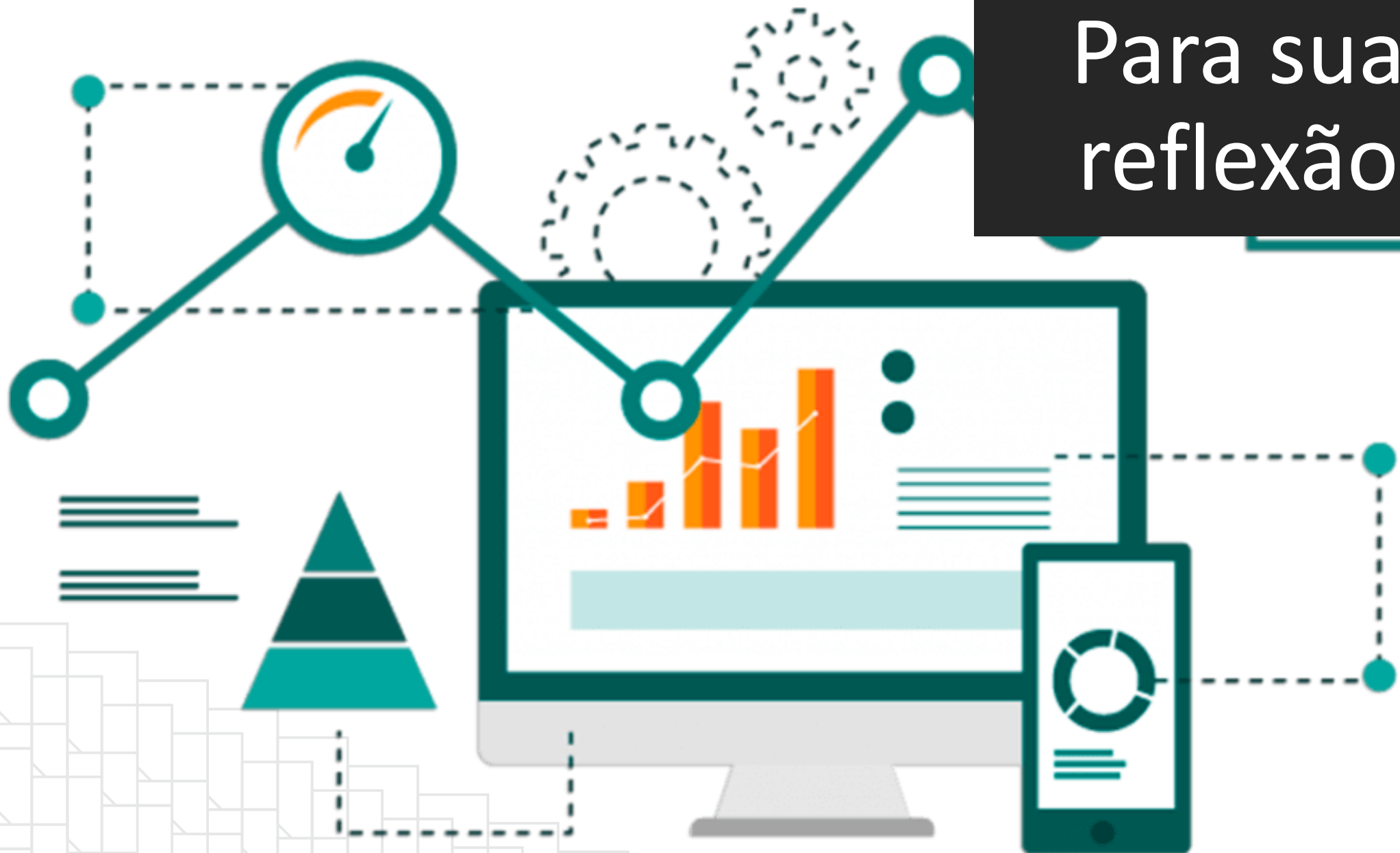
Marcos

SUBMISSÃO # 32447904

PROBLEMA:	2542 - Iu-Di-Oh!
RESPOSTA:	Accepted
LINGUAGEM:	Python 3.8 (Python 3.8.2) [+1s]
TEMPO:	0.294s
TAMANHO:	1,39 KB
MEMÓRIA:	-
SUBMISSÃO:	21/03/2023 17:26:48



Para sua
reflexão



Padawan – Exercício # 1 – Problema 1051

beecrowd | 1051

Imposto de Renda

Por Neilor Tonin, URI  Brasil

Timelimit: 1

Em um país imaginário denominado Lisarb, todos os habitantes ficam felizes em pagar seus impostos, pois sabem que nele não existem políticos corruptos e os recursos arrecadados são utilizados em benefício da população, sem qualquer desvio. A moeda deste país é o Rombus, cujo símbolo é o R\$.

Leia um valor com duas casas decimais, equivalente ao salário de uma pessoa de Lisarb. Em seguida, calcule e mostre o valor que esta pessoa deve pagar de Imposto de Renda, segundo a tabela abaixo.

Renda	Imposto de Renda
de 0.00 a R\$ 2000.00	Isento
de R\$ 2000.01 até R\$ 3000.00	8 %
de R\$ 3000.01 até R\$ 4500.00	18 %
acima de R\$ 4500.00	28 %

Lembre que, se o salário for R\$ 3002.00, a taxa que incide é de 8% apenas sobre R\$ 1000.00, pois a faixa de salário que fica de R\$ 0.00 até R\$ 2000.00 é isenta de Imposto de Renda. No exemplo fornecido (abaixo), a taxa é de 8% sobre R\$ 1000.00 + 18% sobre R\$ 2.00, o que resulta em R\$ 80.36 no total. O valor deve ser impresso com duas casas decimais.

Entrada

A entrada contém apenas um valor de ponto flutuante, com duas casas decimais.



Padawan – Exercício # 1 – Problema 1051

Saída

Imprima o texto "R\$" seguido de um espaço e do valor total devido de Imposto de Renda, com duas casas após o ponto. Se o valor de entrada for menor ou igual a 2000, deverá ser impressa a mensagem "Isento".

Exemplos de Entrada	Exemplos de Saída
3002.00	R\$ 80.36
1701.12	Isento
4520.00	R\$ 355.60



Resolução 1

Python

Exemplos de Saída

R\$ 80.36

Isento

R\$ 355.60

SUBMISSÃO 32334121

LINGUAGEM

PYTHON 3.9

RESPOSTA

Accepted

RUNTIME

0.085s

CÓDIGO FONTE

```
1 salary = float(input())
2
3 if salary <= 2000:
4     print("Isento")
5 else:
6     if salary <= 3000:
7         taxes = (salary - 2000) * 0.08
8     elif salary <= 4500:
9         taxes = 1000 * 0.08 + (salary - 3000) * 0.18
10    else:
11        taxes = 1000 * 0.08 + 1500 * 0.18 + (salary - 4500) * 0.28
12
13    print(f"R$ {taxes:.2f}")
```



Resolução 2

C++

Exemplos de Saída

R\$ 80.36

Isento

R\$ 355.60



SUBMISSÃO 32309719

LINGUAGEM

C++17

RESPOSTA

Accepted

RUNTIME

0.000s

CÓDIGO FONTE

```
1  #include <bits/stdc++.h>
2
3  using namespace std;
4
5  int main(){
6      float valor;
7
8      cin >> valor;
9
10     cout.precision(2);
11     cout.setf(ios::fixed);
12
13     if(valor > 2000){
14         if(valor > 4500){
15             cout << "R$ " << (valor-4500)*(0.28) + 1500*(0.18) + 1000*(0.08) << endl;
16         }else if(valor > 3000){
17             cout << "R$ " << (valor-3000)*(0.18) + 1000*0.08 << endl;
18         }else{
19             cout << "R$ " << (valor-2000)*(0.08) << endl;
20         }
21     }else{
22         cout << "Isento" << endl;
23     }
24
25     return 0;
26
27 }
```

Padawan – Exercício # 2 – Problema 1430

beecrowd | 1430



Composição de Jingles

Por Ines Kerekí Uruguai

Timelimit: 2

A. C. Marcos está dando os primeiros passos para ser um compositor de jingles. Ele está tendo alguns problemas, mas ao menos ele está criando melodias agradáveis e ritmos atrativos.

Na música, uma nota tem um tom (sua frequência, resultando em quão grave ou agudo é o som) e uma duração (por quanto tempo a nota soa). Neste problema, estamos interessados apenas na duração das notas.

Um jingle é dividido em uma sequência de compassos, e um compasso é formado de uma série de notas.

A duração de uma nota é indicada pela sua forma. Neste problema, iremos utilizar letras maiúsculas para indicar a duração de uma nota. A seguinte tabela lista todas as notas disponíveis:

Notas							
Identificador	W	H	Q	E	S	T	X
Duração	1	1/2	1/4	1/8	1/16	1/32	1/64

A duração de um compasso é a soma da duração de suas notas. Nos jingles de Marcos, cada compasso tem a mesma duração. Como Marcos é apenas um iniciante, seu famoso professor Johann Sebastian III o ensinou que a duração de um compasso deve ser sempre 1.

Por exemplo, Marcos escreveu uma composição contendo cinco compassos, dentre os quais quatro possuem a duração correta e um está errado. No exemplo abaixo, cada compasso é delimitado com barras e cada nota é representada como na tabela acima.

/HH/QQQQ/XXXTXTEQH/W/HW/

Marcos gosta de computadores assim como de música. Ele quer que você escreva um programa que determine, para cada uma de suas composições, quantos compassos possuem a duração correta.



Padawan – Exercício # 2 – Problema 1430

Entrada

A entrada contém vários casos de teste. Cada caso de teste é descrito em uma única linha contendo uma string cujo tamanho está entre 3 e 200 caracteres, inclusive, representando uma composição. Uma composição começa e termina com uma barra '/'. Compassos em uma composição são separados por uma barra '/'. Cada nota em um compasso é representada pela letra correspondente segundo a descrição acima. Você pode assumir que cada composição contém ao menos um compasso e que cada compasso contém ao menos uma nota. Todos os caracteres na entrada serão barras ou uma das sete letras maiúsculas usadas para representar as notas.

O último caso de teste é seguido por uma linha contendo um único asterisco.

Saída

Para cada caso de teste, seu programa deve imprimir uma única linha contendo um único inteiro, o número de compassos que possuem a duração correta.

Exemplo de Entrada	Exemplo de Saída
/HH/QQQQ/XXXTXTEQH/W/HW/	4
/W/W/SQHES/	3
/WE/TEX/THES/	0
*	

ACM/ICPC South America Contest 2009.



Resolução 1

C

Exemplo de Saída

4
3
0



SUBMISSÃO 32335622

LINGUAGEM

C99

RESPOSTA

Accepted

RUNTIME

0.001s

CÓDIGO FONTE

```
1  #include <stdio.h>
2  #include <string.h>
3  int main()
4  {
5      char str[1000];
6      int cnt=0, i;
7      double drtn=0.0;
8      while(gets(str))
9      {
10         if(str[0]=='*') break;
11         for(i=0, cnt=0; str[i]; i++)
12         {
13             if(str[i]=='/')
14             {
15                 if(drtn==1.00) cnt++;
16                 drtn = 0.00;
17             }
18             else if(str[i]=='W') drtn+=(1.00);
19             else if(str[i]=='H') drtn+=(1.00/2.0);
20             else if(str[i]=='Q') drtn+=(1.00/4.0);
21             else if(str[i]=='E') drtn+=(1.00/8.0);
22             else if(str[i]=='S') drtn+=(1.00/16.0);
23             else if(str[i]=='T') drtn+=(1.00/32.0);
24             else if(str[i]=='X') drtn+=(1.00/64.0);
25         }
26         printf("%d\n", cnt);
27     }
28     return 0;
29 }
```

Resolução 2 Python

SUBMISSÃO 32374709

NOVO COMENTÁRIO

LINGUAGEM

PYTHON 3.9

RESPOSTA

Accepted

RUNTIME

0.738s

CÓDIGO FONTE

Exemplo de Saída

4
3
0

```
1 def sumStr(bar):
2     return sum([int(note) for note in bar])
3
4 def main():
5     inp = input()
6     while inp != "*":
7         print(sum([1 for summed in [sumStr(x) for x in [x.split("n")[:-1] for x in inp[1:-1].replace("W",
8             "64n").replace("H", "32n").replace("Q", "16n").replace("E", "8n").replace("S", "4n").replace("T", "2n").replace
9             ("X", "1n").split("/")] if summed == 64]]))
10    inp = input()
11    main()
```



Cavaleiro Jedi – Exercício # 1 – Problema 1548

beecrowd | 1548

Fila do Recreio

Por Cristhian Bonilha, UTFPR  Brazil

Timelimit: 1

Na escola onde você estuda, a hora do recreio é a mais aguardada pela grande maioria dos alunos. Não só porque as vezes as aulas são cansativas, mas sim porque a merenda servida é muito boa, preparada por um chefe italiano muito caprichoso.

Quando bate o sinal para a hora do recreio, todos os alunos saem correndo da sua sala para chegar o mais cedo possível na cantina, tanta é a vontade de comer. Um de seus professores notou, porém, que havia ali uma oportunidade.

Utilizando um sistema de recompensa, seu professor de matemática disse que a ordem da fila para se servir será dada não pela ordem de chegada, mas sim pela soma das notas obtidas em sala de aula. Assim, aqueles com maior nota poderão se servir antes daqueles que tem menor nota.

Sua tarefa é simples: dada a ordem de chegada dos alunos na cantina, e as suas respectivas notas na matéria de matemática, reordene a fila de acordo com as notas de matemática, e diga quantos alunos não precisaram trocar de lugar nessa reordenação.



Cavaleiro Jedi – Exercício # 1 – Problema 1548

Entrada

A primeira linha contém um inteiro N , indicando o número de casos de teste a seguir.

Cada caso de teste inicia com um inteiro M ($1 \leq M \leq 1000$), indicando o número de alunos. Em seguida haverá M inteiros distintos P_i ($1 \leq P_i \leq 1000$), onde o i -ésimo inteiro indica a nota do i -ésimo aluno.

Os inteiros acima são dados em ordem de chegada, ou seja, o primeiro inteiro diz respeito ao primeiro aluno a chegar na fila, o segundo inteiro diz respeito ao segundo aluno, e assim sucessivamente.

Saída

Para cada caso de teste imprima uma linha, contendo um inteiro, indicando o número de alunos que não precisaram trocar de lugar mesmo após a fila ser reordenada.

Exemplo de Entrada

```
3
3
100 80 90
4
100 120 30 50
4
100 90 30 25
```

Exemplo de Saída

```
1
0
4
```



Resolução 1

Python

Exemplo de Saída

```
1
0
4
```

SUBMISSÃO 32330418

LINGUAGEM

PYTHON 3

RESPOSTA

Accepted

RUNTIME

0.104s

CÓDIGO FONTE

```
1  n = int(input())
2  for _ in range(n):
3      m = int(input())
4      notas = list(map(int, input().split()))
5      alunos = [(nota, i) for i, nota in enumerate(notas)]
6      alunos.sort(reverse=True)
7      count = sum(alunos[i][1] == i for i in range(m))
8      print(count)
9
```



Resolução 2

C++

Exemplo de Saída

1
0
4



SUBMISSÃO 32310643

LINGUAGEM C++17
RESPOSTA Accepted
RUNTIME 0.033s

CÓDIGO FONTE

```
1  #include <bits/stdc++.h>
2  #define MAXN 1007
3
4  using namespace std;
5
6  int comp(int a, int b){
7      return a > b;
8  }
9
10 int main(){
11     int n;
12     cin >> n;
13
14     for(int i = 0; i < n; i++){
15         int m, fila[MAXN], new_fila[MAXN];
16
17         cin >> m;
18
19         for(int j = 0; j < m; j++){
20             cin >> fila[j];
21             new_fila[j] = fila[j];
22         }
23
24         sort(new_fila, new_fila+m, comp);
25
26         int alunos = 0;
27
28         for(int j = 0; j < m; j++){
29             if(fila[j] == new_fila[j]) alunos++;
30         }
31
32         cout << alunos << endl;
33     }
34 }
35
36 return 0;
37
38 }
```

Cavaleiro Jedi – Exercício # 2 – Problema 1763

beecloud | 1763

Tradutor do Papai Noel
Por Jean Bez, beecloud 🇧🇷 Brazil
Timelimit: 3

Nicolau já está bastante cansado e sua memória não é mais a mesma. Você, como navegador, deverá auxiliar o Papai Noel a gritar a frase "Feliz Natal" no idioma correto de cada país de que trenó está sobrevoando.

Como você é um elfo muito esperto, você já criou um pequeno app no seu celular (sim, elfos tem celular) que irá lhe informar a frase no idioma correto dado o nome do país. Como o trenó é moderno (foi atualizado no ano 2000) ele exibe no painel de navegação o nome do país atual.



Os dados inseridos no seu app foram:

brasil	Feliz Natal!
alemanha	Froehliche Weihnachten!
austria	Frohe Weihnacht!
coreia	Chuk Sung Tan!
espanha	Feliz Navidad!
greCIA	Kala Christougena!
estados-unidos	Merry Christmas!
inglaterra	Merry Christmas!
australia	Merry Christmas!
portugal	Feliz Natal!
suecia	God Jul!
turquia	Mutlu Noeller
argentina	Feliz Navidad!
chile	Feliz Navidad!
mexico	Feliz Navidad!
antardida	Merry Christmas!
canada	Merry Christmas!
irlanda	Nollaig Shona Dhuit!
belgica	Zalig Kerstfeest!
italia	Buon Natale!
libia	Buon Natale!
siria	Milad Mubarak!
marrocos	Milad Mubarak!
japao	Merii Kurisumasu!

Para não correr o risco de infomar o nome errado você decidiu testar o aplicativo mais algumas vezes.

Cavaleiro Jedi – Exercício # 2 – Problema 1763

Entrada

Você irá testar o seu aplicativo com diversos nomes de países, simulando os dados informados pelo painel de navegação do trenó.

Saída

O seu aplicativo deverá mostrar na tela a frase no idioma correto. Caso ela não esteja cadastrada, você deverá exibir a mensagem "*--- NOT FOUND ---*" para que depois dos testes você possa completar o banco de dados.

Exemplo de Entrada	Exemplo de Saída
<code>uri-online-judge</code> <code>alemanha</code> <code>brasil</code> <code>austria</code>	<code>--- NOT FOUND ---</code> <code>Frohliche Weihnachten!</code> <code>Feliz Natal!</code> <code>Frohe Weihnacht!</code>

Contest de Natal 2014



Resolução 1

Python

LINGUAGEM

PYTHON 3.9

RESPOSTA

Accepted

RUNTIME

0.645s

CÓDIGO FONTE

Exemplo de Saída

```
--- NOT FOUND ---  
Frohliche Weihnachten!  
Feliz Natal!  
Frohe Weihnacht!
```

```
1 def func(entrada):  
2     paises = ['brasil', 'alemanha', 'austria', 'coreia', 'espanha', 'grecia', 'estados-unidos', 'inglaterra',  
               'australia', 'portugal', 'suecia', 'turquia', 'argentina', 'chile', 'mexico', 'antardida', 'canada',  
               'irlanda', 'belgica', 'italia', 'libia', 'siria', 'marrocos', 'japao']  
3     frases = ['Feliz Natal!', 'Frohliche Weihnachten!', 'Frohe Weihnacht!', 'Chuk Sung Tan!', 'Feliz Navidad!',  
               'Kala Christougena!', 'Merry Christmas!', 'Merry Christmas!', 'Merry Christmas!', 'Feliz Natal!', 'God  
               Jul!', 'Mutlu Noeller', 'Feliz Navidad!', 'Feliz Navidad!', 'Feliz Navidad!', 'Merry Christmas!', 'Merry  
               Christmas!', 'Nollaig Shona Dhuit!', 'Zalig Kerstfeest!', 'Buon Natale!', 'Buon Natale!', 'Milad Mubarak!',  
               'Milad Mubarak!', 'Merii Kurisumasu!']  
4  
5     for i in range(len(paises)):  
6         if paises[i] == entrada:  
7             print(frases[i])  
8             return  
9     print("--- NOT FOUND ---")  
10  
11 vet_entradas = []  
12  
13 while True:  
14     try:  
15         entrada = input()  
16         vet_entradas.append(entrada)  
17     except:  
18         break  
19  
20 for i in vet_entradas:  
21     func(i)
```



Cavaleiro Jedi - Exercício #2 - Problema 1763

Resolução 1

C++

Exemplo de Saída

```
--- NOT FOUND ---
Frohliche Weihnachten!
Feliz Natal!
Frohe Weihnacht!
```

LINGUAGEM

C++17

RESPOSTA

Accepted

RUNTIME

0.606s

CÓDIGO FONTE

```
1  #include <bits/stdc++.h>
2
3  using namespace std;
4
5  int main() {
6      string frase;
7
8      string ptbr[] = {"brasil", "alemanha", "austria", "coreia", "espanha", "grecia", "estados-unidos",
9                      "inglaterra", "australia", "portugal", "suecia", "turquia",
10                     "argentina", "chile", "mexico", "antardida",
11                     "canada", "irlanda", "belgica", "italia",
12                     "libia", "siria", "marrocos", "japao"};
13
14     string feliznatal[] = {"Feliz Natal!", "Frohliche Weihnachten!",
15                           "Frohe Weihnacht!", "Chuk Sung Tan!",
16                           "Feliz Navidad!", "Kala Christougena!", "Merry Christmas!", "Merry Christmas!",
17                           "Merry Christmas!", "Feliz Natal!",
18                           "God Jul!", "Mutlu Noeller",
19                           "Feliz Navidad!", "Feliz Navidad!", "Feliz Navidad!", "Merry Christmas!", "Merry Christmas!", "Nollaig
20                           Shona Dhuit!", "Zalig Kerstfeest!", "Buon Natale!", "Buon Natale!", "Milad Mubarak!", "Milad Mubarak!",
21                           "Merii Kurisumasu!"};
22
23     map<string, string> dic;
24
25     for (int i = 0; i < 24; i++) {
26         dic.insert({ptbr[i], feliznatal[i]});
27     }
28
29     while (cin >> frase) {
30         if(dic.find(frase) != dic.end()){
31             cout << dic[frase] << endl;
32         }else{
33             cout << "--- NOT FOUND ---" << endl;
34         }
35     }
36
37     return 0;
38 }
```



Mestre Yoda – Exercício # 1 – Problema 2370

beecrowd | 2370

Times

Por OBI - Olimpíada Brasileira de Informática 2010 🇧🇷 Brazil

Timelimit: 1

As aulas de educação física, em muitas escolas, acontecem da seguinte maneira: O professor entrega uma bola ao alunos (geralmente de futebol) e estes se dividem em times, onde jogam partidas alternadamente.

A maneira como os times são escolhidos também é semelhante em todas as escolas: decide-se quantos times serão formados, e uma pessoa para montar cada um dos times. Cada pessoa vai escolher, alternadamente, um dos alunos restantes para fazer parte de sua equipe. Como todos querem ter uma boa equipe, a pessoa que vai escolher o próximo membro do time escolhe aquele, dentre os ainda disponíveis, que possui o melhor nível de habilidade. Assim, os times acabam ficando relativamente equilibrados na soma do nível de habilidade dos jogadores.

Dada uma lista de alunos que serão escolhidos e seus respectivos níveis de habilidade para os times e a quantidade de times que serão formados, mostre como ficarão os times ao final do processo de montagem dos mesmos.

Entrada

A primeira linha da entrada contém dois inteiros N ($2 \leq N \leq 10.000$) e T ($2 \leq T \leq 1000$), representando respectivamente a quantidade de alunos e o número de times a serem formados, sendo $T \leq N$. As N linhas seguintes descrevem, cada uma, um aluno disponível para escolha de times. Cada uma dessas linhas possui o nome do aluno (composto apenas por letras minúsculas) e um inteiro H ($0 \leq H \leq 1.000.000$) descrevendo seu nível de habilidade).

Não existem dois alunos com o mesmo nível de habilidade, e todos eles possuem nomes diferentes. É possível que alguns times acabem ficando com menos jogadores do que os outros.

Saída

Seu programa deve imprimir a lista de times que será formada ao final do processo de seleção. Para cada time, você deverá mostrar o termo "Time N", onde N é o número do time (1 para o primeiro, 2 para o segundo, e assim por diante) seguido de K linhas, onde K é a quantidade de jogadores do time, mostrando o nome de cada um dos jogadores do time, em ordem alfabética. Imprima uma linha em branco após cada descrição de time (inclusive do último).

Os times serão escolhidos pelo computador, então não é necessário considerar o aluno que irá fazer a escolha dos times.

Mestre Yoda – Exercício # 1 – Problema 2370

Exemplo de Entrada

```
8
1 3 5 7 11 13 17 19
```

Exemplo de Saída

```
8 68
```

```
8
1 3 5 7 11 13 16 19
```

```
7 63
```

3ª Maratona de Programação Júnior da UFFS



Mestre Yoda - Exercício #1 - Problema 2370

Resolução 1

Python

Exemplo de Saída

8 68

7 63

SUBMISSÃO 32329243

LINGUAGEM

PYTHON 3.9

RESPOSTA

Accepted

RUNTIME

0.080s

CÓDIGO FONTE

```
1  # -*- coding: utf-8 -*-
2
3  '''
4  Escreva a sua solução aqui
5  Code your solution here
6  Escriba su solución aquí
7  '''
8  def main():
9      student_qty, team_qty = map(int, input().split())
10
11      students = sorted([input().split() for i in range(student_qty)], key=lambda item: int(item[1]))
12
13      teams = [[] for i in range(team_qty)]
14
15      while len(students) > 0:
16          for i in range(team_qty):
17              if len(students) == 0:
18                  break
19
20              teams[i].append(students.pop(-1))
21
22      teams = [sorted(team, key=lambda item: item[0]) for team in teams]
23
24      for index, team in enumerate(teams):
25          print(f'Time {index+1}')
26          for student in team:
27              print(student[0])
28          print()
29
30  main()
```



Resolução 2

C++

Exemplo de Saída

8 68

7 63

LINGUAGEM

C++17

RESPOSTA

Accepted

RUNTIME

0.051s

CÓDIGO FONTE

```

1  #include <bits/stdc++.h>
2  #define MAXN 1007
3
4  using namespace std;
5
6  int main(){
7      int qtde_a, qtde_times;
8
9      vector <string> times[MAXN];
10     priority_queue <pair <int, string>, vector <pair <int, string>>> fila;
11
12     cin >> qtde_a >> qtde_times;
13
14     for(int i = 0; i < qtde_a; i++){
15         string nome;
16         int hab;
17         cin >> nome >> hab;
18         fila.push(make_pair(hab, nome));
19     }
20
21     for(int i = 0; i < qtde_a; i++){
22         string melhor = fila.top().second;
23         fila.pop();
24         times[i%qtde_times].push_back(melhor);
25     }
26
27     for(int i = 0; i < qtde_times; i++){
28         sort(times[i].begin(), times[i].end());
29     }
30
31     for(int i = 0; i < qtde_times; i++){
32         cout << "Time " << i+1 << endl;
33         for(int j = 0; j < times[i].size(); j++){
34             cout << times[i][j] << endl;
35         }
36         cout << endl;
37     }
38
39     return 0;
40
41 }
```



Mestre Yoda – Exercício # 2 – Problema 1069



beecrowd | 1069

Diamantes e Areia

Por Nello Tonin, URI  Brasil

Timelimit: 1

João está trabalhando em uma mina, tentando retirar o máximo que consegue de diamantes "<>". Ele deve excluir todas as partículas de areia "." do processo e a cada retirada de diamante, novos diamantes poderão se formar. Se ele tem como uma entrada .<...<.>...>...>>., três diamantes são formados. O primeiro é retirado de <...>, resultando .<...<.>...>...>>.. Em seguida o segundo diamante é retirado, restando .<.....>...>>.. O terceiro diamante é então retirado, restando no final>>., sem possibilidade de extração de novo diamante.

Entrada

Deve ser lido um valor inteiro **N** que representa a quantidade de casos de teste. Cada linha a seguir é um caso de teste que contém até 1000 caracteres, incluindo "<, >, ."

Saída

Você deve imprimir a quantidade de diamantes possíveis de serem extraídos em cada caso de entrada.

Exemplo de Entrada	Exemplo de Saída
<pre>2 <...><.<.>...> <<<.<.....<<<<.....></pre>	<pre>3 1</pre>



Resolução 1

Python

Exemplo de Saída

```
3
1
```

SUBMISSÃO 32320792

LINGUAGEM

PYTHON 3.9

RESPOSTA

Accepted

RUNTIME

0.039s

CÓDIGO FONTE

```
1  reps = int(input())
2  for i in range(reps):
3      mine = input().replace(".", "")
4      count = 0
5      while True:
6          try:
7              mine.index("<>")
8              mine = mine.replace("<>", "", 1)
9              count += 1
10         except:
11             break
12     print(count)
```



Resolução 2

C++

Exemplo de Saída

```
3
1
```



SUBMISSÃO 32321340

LINGUAGEM

C++17

RESPOSTA

Accepted

RUNTIME

0.000s

CÓDIGO FONTE

```
1  #include <bits/stdc++.h>
2
3  using namespace std;
4
5  int main(){
6      int n;
7      cin >> n;
8
9      for(int i = 0; i < n; i++){
10         string mina;
11         int cont = 0;
12         stack <char> diam;
13
14         cin >> mina;
15
16         for(int i = 0; i < mina.size(); i++){
17             if(mina[i] == '<') diam.push(mina[i]);
18             else if(mina[i] == '>'){
19                 if(diam.size() != 0){
20                     diam.pop();
21                     cont++;
22                 }
23             }
24         }
25
26         cout << cont << endl;
27     }
28
29     return 0;
30
31 }
32 }
```

Responder o formulário a seguir para contabilizar a sua presença:

- <https://forms.office.com/r/d1F9iLsH5v>



Quiz #3 - 2023/1 - Treinamento Maratona de Programação

Enquete aplicada aos participantes do segundo encontro do Treinamento Maratona de Programação do 1o semestre de 2023

🕒 5 minutos

Olá IVAN, este é um formulário cronometrado.

Depois de iniciar, você não poderá pausar o timer. Não se preocupe, o Forms fornece um lembrete de minuto final antes do envio. Suas respostas serão enviadas automaticamente quando o tempo expirar. Prepare-se antes de começar a ajudar a gerenciar seu tempo de envio.



Dúvidas!?

Hands On



