

# Treinamento Maratona de Programação

*Extensão – 2o Encontro*

Prof. Dr. Alexandre dos Santos Mignon ([alexandre.mignon@mackenzie.br](mailto:alexandre.mignon@mackenzie.br))

Prof. Ms. Fabio Lubacheski ([fabio.lubacheski@mackenzie.br](mailto:fabio.lubacheski@mackenzie.br))

Prof. Dr. Ivan Carlos Alcântara de Oliveira ([ivan.oliveira@mackenzie.br](mailto:ivan.oliveira@mackenzie.br))

Prof. Dr. Leandro Carlos Fernandes ([leandro.fernandes@mackenzie.br](mailto:leandro.fernandes@mackenzie.br))



# Objetivos



Nesta aula temos como meta:

- Alterações no planejamento inicial
- Feedback da plataforma: mensagens resultantes da correção
- Comentários sobre soluções submetidas (Exercícios #1 e #2)
- Estudo de Caso: Problema das Moedas
- Esclarecimento de dúvidas em geral e considerações finais



## Calendário dos Encontros do Treinamento Maratona de Programação

| Encontro<br>(Evento) | 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.

# Tipos de feedback da plataforma

**Compilation Error** (o programa não compilou): verificar se a linguagem de programação foi escolhida corretamente e possíveis erros.

**Run time error** (o programa compilou corretamente, entretanto teve um erro em tempo de execução): avaliar a solução na busca de possíveis erros de programação que originem acessos inválidos de memória, estouros de pilha, falhas de segmentação etc.

**Wrong Answer** (o programa compilou e executou sem erros, entretanto não forneceu a saída correta para todos os casos): investigar se o algoritmo proposto funciona para todos os casos, revisar a resolução do problema na busca de possíveis condições não detectadas anteriormente.



# Tipos de feedback da plataforma

**Presentation Error** (o programa compilou e executou sem erros, além disso forneceu a resposta correta, entretanto possui algum erro de formatação da saída): analisar se a saída está no formato especificado pela questão.

**Time Limit Exceeded** (o programa demorou mais para terminar do que o permitido pelos juízes): avaliar a possibilidade do programa não estar parando em determinados casos. Deve-se verificar se o algoritmo proposto é rápido o suficiente para o problema proposto.

**Accepted** (o programa compilou e executou sem erros, além disso forneceu a resposta correta para todos os casos teste): o programa está correto, um balão colorido está a caminho...



# Tipos de feedback da plataforma

**IN QUEUE.** A sua submissão está na fila para ser julgada.

**THINKING...** A submissão está sendo re-julgada devido a casos de teste ou especificação incorreta..

**CLOSED.** Houve um problema com esta submissão. Provavelmente não recebemos o código-fonte, por isso ela foi finalizada.

**MEMORY LIMIT EXCEEDED.** Se a sua solução recebeu esta resposta, significa que seu código tentou alocar mais memória do que o máximo permitido para o problema. Isso pode ocorrer porque você está utilizando um vetor ou uma estrutura de dados muito grande.



# Tipos de feedback da plataforma

***POSSIBLE RUNTIME ERROR.*** Se a sua solução recebeu essa mensagem um dos seguintes problemas pode ter acontecido:

- Sua classe em Java não foi nomeada corretamente. Ele sempre deve ser nomeada como Main;
- Você está utilizando pacotes;
- Você está utilizando bibliotecas gráficas, como o Swing, ou bibliotecas de rede;



# Tipos de feedback da plataforma

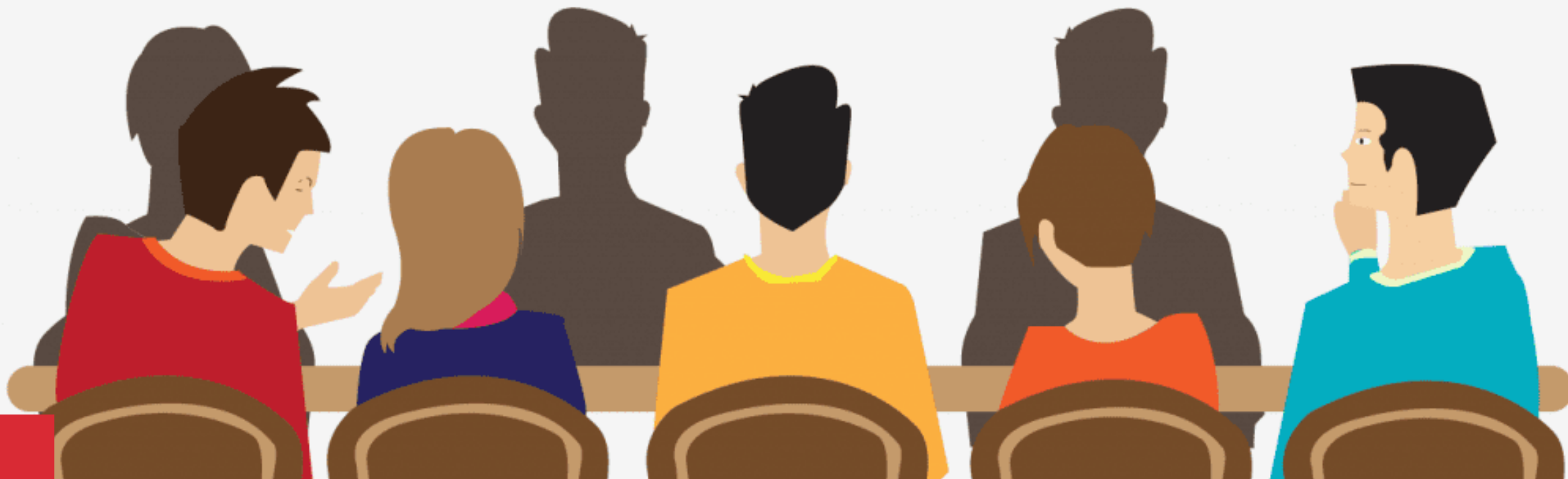
***POSSIBLE RUNTIME ERROR.*** Se a sua solução recebeu essa mensagem um dos seguintes problemas pode ter acontecido:

- Você não está lendo a entrada conforme a descrição do problema. Por exemplo, se o problema informar que a linha seguinte contém vários números e você os ler como se estivessem apresentados um por linha, provavelmente seu código lançará uma exceção e você receberá como resposta "*Possible Runtime Error*";
- Você não definiu o *Locale* padrão e tentou ler valores de moeda com vírgula ao invés de ponto. Lembre-se sempre de setar o *Locale* na primeira linha do método main.





# Discussão



# Padawan - Exercício #1 - Problema 1017




BEE 1017

INICIANTE | NÍVEL 1 | + 1.9 PONTOS | TEMPO LIMITE BASE: 1 SEGUNDOS | LIMITE DE MEMÓRIA: 200 MB

beecrowd | 1017



## Gasto de Combustível

Adaptado por Neilor Tonin, URI  Brasil

**Timelimit: 1**

Joaozinho quer calcular e mostrar a quantidade de litros de combustível gastos em uma viagem, ao utilizar um automóvel que faz 12 KM/L. Para isso, ele gostaria que você o auxiliasse através de um simples programa. Para efetuar o cálculo, deve-se fornecer o tempo gasto na viagem (em horas) e a velocidade média durante a mesma (em km/h). Assim, pode-se obter distância percorrida e, em seguida, calcular quantos litros seriam necessários. Mostre o valor com 3 casas decimais após o ponto.

### Entrada

O arquivo de entrada contém dois inteiros. O primeiro é o tempo gasto na viagem (em horas) e o segundo é a velocidade média durante a mesma (em km/h).



# Padawan - Exercício #1 - Problema 1017

## Entrada

O arquivo de entrada contém dois inteiros. O primeiro é o tempo gasto na viagem (em horas) e o segundo é a velocidade média durante a mesma (em km/h).

## Saída

Imprima a quantidade de litros necessária para realizar a viagem, com três dígitos após o ponto decimal

| Exemplo de Entrada | Exemplo de Saída |
|--------------------|------------------|
| 10<br>85           | 70.833           |
| 2<br>92            | 15.333           |
| 22<br>67           | 122.833          |



# Resolução 1 (Python)

## Exemplo de Saída

70.833

15.333

122.833

## SUBMISSÃO 32199367

LINGUAGEM

PYTHON 3.9

RESPOSTA

Wrong answer (10%)

RUNTIME

0.017s

## CÓDIGO FONTE

```
1 time = int(input( ))
2 kmh = int(input( ))
3 gasoline = round((kmh*time)/12, 3)
4 print (f'{gasoline}')
```



# Resolução 2 (Python)

## Exemplo de Saída

70.833

15.333

122.833

SUBMISSÃO 32173731

LINGUAGEM

PYTHON 3

RESPOSTA

Wrong answer (45%)

RUNTIME

0.034s

## CÓDIGO FONTE

```
1 Tempo =int(input())
2 VelocidadeMedia= int(input())
3
4 Kmrodado = Tempo*VelocidadeMedia
5 Litros= Kmrodado/12
6 print(Litros)
```



# Resolução 3 (Python)

LINGUAGEM

PYTHON 3

RESPOSTA

Wrong answer (90%)

RUNTIME

0.015s

## CÓDIGO FONTE

```
1 spent_time = int(input("How long did your trip take (in hours): "))
2 average_speed = int(input("What was your average speed (in Km/h): "))
3
4 distance = spent_time * average_speed
5 liters = 0
6 float(liters)
7 liters = distance/12
8
9 print("Amount of liters needed: %.3f" %liters)
```

### Exemplo de Saída

70.833

15.333

122.833



# Resolução 3 (C)

LINGUAGEM

C99

RESPOSTA

Presentation error (100%)

RUNTIME

0.000s

## CÓDIGO FONTE

### Exemplo

70.833

15.333

122.833

```
1  #include <stdio.h>
2
3  int main() {
4
5      int velocidade, tempo;
6      scanf("%d", &tempo);
7      scanf("%d", &velocidade);
8      float distancia = velocidade*tempo;
9      float litros = distancia/12;
10     printf("%.3f", litros);
11
12     return 0;
13 }
```

SUBMISSÃO 32191563

INGUAGEM

C99

ESPOSTA

Accepted

UNTIME

0.000s

## CÓDIGO FONTE

```
1  #include <stdio.h>
2
3  int main() {
4
5      int velocidade, tempo;
6      scanf("%d", &tempo);
7      scanf("%d", &velocidade);
8      float distancia = velocidade*tempo;
9      float litros = distancia/12;
10     printf("%.3f\n", litros);
11
12     return 0;
13 }
```



# Resolução 4 (Java - 1)

## Exemplo de Saída

70.833

15.333

122.833



SUBMISSÃO 32213745

LINGUAGEM

JAVA 14

RESPOSTA

Presentation error (100%)

RUNTIME

0.273s

## CÓDIGO FONTE

```
1  import java.io.IOException;
2  import java.util.Scanner;
3
4  public class Main {
5      public static void main(String[] args) throws IOException {
6          double tempGasto;
7          double velMedia;
8          double litroNecessario;
9          Scanner entrada = new Scanner(System.in);
10         /*Tempo gasto*/System.out.println();
11         tempGasto = entrada.nextDouble();
12         /*velocidade media*/System.out.println();
13         velMedia = entrada.nextDouble();
14         litroNecessario = (velMedia*tempGasto)/12;
15         System.out.printf("%.3f", litroNecessario);
16         System.out.println();
17         entrada.close();
18         System.exit(0);
19     }
20 }
21
```



# Resolução 4 (Java - 2)

## Exemplo de Saída

70.833

15.333

122.833



SUBMISSÃO 32213832

LINGUAGEM

JAVA 14

RESPOSTA

Compilation error

RUNTIME

0.000s

## CÓDIGO FONTE

Deveria ser, talvez:  
`System.out.print(false);`

```
1 import java.io.IOException;
2 import java.util.Scanner;
3
4 public class Main {
5     public static void main(String[] args) throws IOException {
6         double tempGasto;
7         double velMedia;
8         double litroNecessario;
9         Scanner entrada = new Scanner(System.in);
10        /*Tempo gasto*/System.out.print();
11        tempGasto = entrada.nextDouble();
12        /*velocidade media*/System.out.print();
13        velMedia = entrada.nextDouble();
14        litroNecessario = (velMedia*tempGasto)/12;
15        System.out.printf("%.3f", litroNecessario);
16
17        entrada.close();
18        System.exit(0);
19    }
20 }
21
22
```

# Resolução 4 (Java - 3)

## Exemplo de Saída

70.833

15.333

122.833



SUBMISSÃO 32213848

LINGUAGEM

JAVA 14

RESPOSTA

Accepted

RUNTIME

0.341s

## CÓDIGO FONTE

```
1  import java.io.IOException;
2  import java.util.Scanner;
3
4
5  public class Main {
6      public static void main(String[] args) throws IOException {
7          double tempGasto;
8          double velMedia;
9          double litroNecessario;
10         Scanner entrada = new Scanner(System.in);
11         /*Tempo gasto*/
12         tempGasto = entrada.nextDouble();
13         /*velocidade media*/
14         velMedia = entrada.nextDouble();
15         litroNecessario = (velMedia*tempGasto)/12;
16         System.out.printf("%.3f", litroNecessario);
17         System.out.println();
18         entrada.close();
19         System.exit(0);
20     }
21
22 }
23
```

# SUBMISSÃO 32187138

LINGUAGEM

PYTHON 3

RESPOSTA

Runtime error

RUNTIME

0.177s

## Runtime Error

## CÓDIGO FONTE

```
1 x, y = input().split()
2 out = x*y/12
3 print('{0:.3f}'.format(out))
```

Faltou converter  
x, y para float



# Resolução 6 (C++)

## Exemplo de Saída

70.833

15.333

122.833



SUBMISSÃO 32171422

LINGUAGEM

C++17

RESPOSTA

Wrong answer (30%)

RUNTIME

0.000s

## CÓDIGO FONTE


```
1  #include <iostream>
2  #include <iomanip>
3
4  using namespace std;
5
6  int main() {
7      const int mediaGas = 12;
8      int tempo, velMedia = 0;
9      double litrosNecessarios;
10
11     cin >> tempo >> velMedia;
12     litrosNecessarios = (velMedia * tempo)/mediaGas;
13
14     cout << fixed << setprecision(3) << litrosNecessarios << "\n";
15     return 0;
16 }
```

O resultado da divisão será inteiro, pois o divisor e o dividendo são inteiros.

# Estudo de Caso




# *The devil is in the details*

 **BEE 1021**

INICIANTE | **NÍVEL 6** | + 6.3 PONTOS | TEMPO LIMITE BASE: 1 SEGUNDOS | LIMITE DE MEMÓRIA: 200 MB

beecrowd | 1021

**Notas e Moedas**

Por Neilor Tonin, URI  Brasil

**Timelimit: 1**

---

Leia um valor de ponto flutuante com duas casas decimais. Este valor representa um valor monetário. A seguir, calcule o menor número de notas e moedas possíveis no qual o valor pode ser decomposto. As notas consideradas são de 100, 50, 20, 10, 5, 2. As moedas possíveis são de 1, 0.50, 0.25, 0.10, 0.05 e 0.01. A seguir mostre a relação de notas necessárias.

## Entrada

O arquivo de entrada contém um valor de ponto flutuante **N** ( $0 \leq N \leq 1000000.00$ ).

## Saída

Imprima a quantidade mínima de notas e moedas necessárias para trocar o valor inicial, conforme exemplo fornecido.

Obs: Utilize ponto (.) para separar a parte decimal.

| Exemplo de Entrada | Exemplo de Saída   |
|--------------------|--|
| 576.73             | NOTAS:<br>5 nota(s) de R\$ 100.00<br>1 nota(s) de R\$ 50.00<br>1 nota(s) de R\$ 20.00<br>0 nota(s) de R\$ 10.00<br>1 nota(s) de R\$ 5.00<br>0 nota(s) de R\$ 2.00<br>MOEDAS:<br>1 moeda(s) de R\$ 1.00<br>1 moeda(s) de R\$ 0.50<br>0 moeda(s) de R\$ 0.25<br>2 moeda(s) de R\$ 0.10<br>0 moeda(s) de R\$ 0.05<br>3 moeda(s) de R\$ 0.01 |





# Tentativa 1 (Python)

## SUBMISSÃO # 29459336

PROBLEMA: 1021 - Notas e Moedas  
RESPOSTA: **Wrong answer (10%)**  
LINGUAGEM: Python 3.8 (Python 3.8.2) [+1s]  
TEMPO: 0.226s  
TAMANHO: 416 Bytes  
MEMÓRIA: -  
SUBMISSÃO: 25/08/2022 17:07:10

## CÓDIGO FONTE

```
1 # -*- coding: utf-8 -*-
2
3 cedulas = [100, 50, 20, 10, 5, 2]
4 moedas = [1, 0.50, 0.25, 0.10, 0.05, 0.01]
5
6 valor = float(input())
7
8 print("NOTAS:")
9 for c in cedulas:
10     qtde = valor // c
11     valor = valor - (qtde*c)
12     print("%d nota(s) de R$ %.2f" %(qtde, c))
13
14 print("MOEDAS:")
15 for m in moedas:
16     qtde = valor // m
17     valor = valor - (qtde*m)
18     print("%d nota(s) de R$ %.2f" %(qtde, m))
19
```

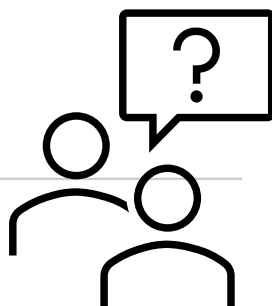




# Tentativa 2 (Python)

## SUBMISSÃO # 29459405

PROBLEMA: 1021 - Notas e Moedas  
RESPOSTA: **Wrong answer (100%)**  
LINGUAGEM: Python 3.8 (Python 3.8.2) [+1s]  
TEMPO: 0.024s  
TAMANHO: 417 Bytes  
MEMÓRIA: -  
SUBMISSÃO: 25/08/2022 17:10:35



## CÓDIGO FONTE

```
1 # -*- coding: utf-8 -*-
2
3 cedulas = [100, 50, 20, 10, 5, 2]
4 moedas = [1, 0.50, 0.25, 0.10, 0.05, 0.01]
5
6 valor = float(input())
7
8 print("NOTAS:")
9 for c in cedulas:
10     qtde = valor // c
11     valor = valor - (qtde*c)
12     print("%d nota(s) de R$ %.2f" %(qtde, c))
13
14 print("MOEDAS:")
15 for m in moedas:
16     qtde = valor // m
17     valor = valor - (qtde*m)
18     print("%d moeda(s) de R$ %.2f" %(qtde, m))
19
```

```
9.99
NOTAS:
0 0.99
0 NOTAS:
0 0 nota(s) de R$ 100.00
0 0 nota(s) de R$ 50.00
1 0 nota(s) de R$ 20.00
2 0 nota(s) de R$ 10.00
0 0 nota(s) de R$ 5.00
0 0 nota(s) de R$ 2.00
1 MOEDAS:
1 0 moeda(s) de R$ 1.00
2 1 moeda(s) de R$ 0.50
0 1 moeda(s) de R$ 0.25
0 4 2 moeda(s) de R$ 0.10
0 moeda(s) de R$ 0.05
3 moeda(s) de R$ 0.01
```

# Tentativa 3 (C/C++)

## SUBMISSÃO # 29460512

|            |   |
|------------|---|
| PROBLEMA:  | 1021 - Notas e Moedas                       |
| RESPOSTA:  | <b>Wrong answer (5%)</b>                    |
| LINGUAGEM: | C++17 (g++ 7.3.0, -std=c++17 -O2 -lm) [+0s] |
| TEMPO:     | 0.000s                                      |
| TAMANHO:   | 705 Bytes                                   |
| MEMÓRIA:   | -   |
| SUBMISSÃO: | 25/08/2022 18:07:54                         |

## CÓDIGO FONTE

```
1  #include <stdio.h>
2
3  int cédulas[] = {100, 50, 20, 10, 5, 2};
4  float moedas[] = {1, 0.5, 0.25, 0.1, 0.05, 0.01};
5
6  int main() {
7      float valor;
8      int qtde;
9      scanf("%f", &valor);
10
11     printf("NOTAS:\n");
12     for(int i=0; i<6;i++) {
13         qtde = 0;
14         while (valor >= cédulas[i]) {
15             qtde++;
16             valor -= cédulas[i];
17         }
18         printf("%d nota(s) de R$ %.2f\n", qtde, cédulas[i]);
19     }
20     printf("MOEDAS:\n");
21     for(int i=0; i<6;i++) {
22         qtde = 0;
23         while (valor >= moedas[i]) {
24             qtde++;
25             valor -= moedas[i];
26         }
27         printf("%d moeda(s) de R$ %.2f\n", qtde, moedas[i]);
28     }
29     return 0;
30 }
```



# Tentativa 4 (C)

## SUBMISSÃO # 29460813

PROBLEMA: 1021 - Notas e Moedas  
RESPOSTA: **Wrong answer (5%)**  
LINGUAGEM: C99 (gcc 4.8.5, -std=c99 -O2 -lm) [+0s]  
TEMPO: 0.000s  
TAMANHO: 706 Bytes  
MEMÓRIA: -  
SUBMISSÃO: 25/08/2022 18:23:00

## CÓDIGO FONTE

```
1  #include <stdio.h>
2
3  int cedulas[] = {100, 50, 20, 10, 5, 2};
4  float moedas[] = {1, 0.5, 0.25, 0.1, 0.05, 0.01};
5
6  int main() {
7      float valor;
8      int qtde;
9      scanf("%f", &valor);
10
11     printf("NOTAS:\n");
12     for(int i=0; i<6;i++) {
13         qtde = 0;
14         while (valor >= cedulas[i]) {
15             qtde++;
16             valor -= cedulas[i];
17         }
18         printf("%d nota(s) de R$ %d.00\n", qtde, cedulas[i]);
19     }
20     printf("MOEDAS:\n");
21     for(int i=0; i<6;i++) {
22         qtde = 0;
23         while (valor >= moedas[i]) {
24             qtde++;
25             valor -= moedas[i];
26         }
27         printf("%d moeda(s) de R$ %.2f\n", qtde, moedas[i]);
28     }
29     return 0;
30 }
```



# Tentativa 5 (C)

## SUBMISSÃO # 29460970

|            |   |
|------------|---|
| PROBLEMA:  | 1021 - Notas e Moedas                   |
| RESPOSTA:  | <b>Wrong answer (5%)</b>                |
| LINGUAGEM: | C99 (gcc 4.8.5, -std=c99 -O2 -lm) [+0s] |
| TEMPO:     | 0.007s                                  |
| TAMANHO:   | 864 Bytes                               |
| MEMÓRIA:   | -                                       |
| SUBMISSÃO: | 25/08/2022 18:31:02                     |



## CÓDIGO FONTE

```
1  #include <stdio.h>
2
3  int cedulas[] = {100, 50, 20, 10, 5, 2};
4  float moedas[] = {1, 0.5, 0.25, 0.1, 0.05, 0.01};
5
6  int main() {
7      float valor;
8      int qtde;
9      scanf("%f", &valor);
10
11     printf("NOTAS:\n");
12     for(int i=0; i<6;i++) {
13         qtde = 0;
14         while (valor >= cedulas[i]) {
15             qtde++;
16             valor -= cedulas[i];
17         }
18         printf("%d nota(s) de R$ %d.00\n", qtde, cedulas[i]);
19     }
20     printf("MOEDAS:\n");
21     for(int i=0; i<5;i++) {
22         qtde = 0;
23         while (valor >= moedas[i]) {
24             qtde++;
25             valor -= moedas[i];
26         }
27         printf("%d moeda(s) de R$ %.2f\n", qtde, moedas[i]);
28     }
29
30     qtde = 0;
31     while (valor >= moedas[5]) {
32         qtde++;
33         valor -= moedas[5];
34     }
35     printf("%d moeda(s) de R$ 0.01", qtde);
36
37     return 0;
38 }
```

# Tentativa 6 (C)

## SUBMISSÃO # 29461075

PROBLEMA: 1021 - Notas e Moedas  
RESPOSTA: **Wrong answer (5%)**  
LINGUAGEM: C99 (gcc 4.8.5, -std=c99 -O2 -lm) [+0s]  
TEMPO: 0.000s  
TAMANHO: 710 Bytes  
MEMÓRIA: -  
SUBMISSÃO: 25/08/2022 18:35:40



## CÓDIGO FONTE

```
1  #include <stdio.h>
2
3  int cédulas[] = {100, 50, 20, 10, 5, 2};
4  float moedas[] = {1, 0.5, 0.25, 0.1, 0.05, 0.01};
5
6  int main() {
7      float valor;
8      int qtde;
9      scanf("%f", &valor);
10
11     printf("NOTAS:");
12     for(int i=0; i<6;i++) {
13         qtde = 0;
14         while (valor >= cédulas[i]) {
15             qtde++;
16             valor -= cédulas[i];
17         }
18         printf("\n%d nota(s) de R$ %d.00", qtde, cédulas[i]);
19     }
20     printf("\nMOEDAS:");
21     for(int i=0; i<6;i++) {
22         qtde = 0;
23         while (valor >= moedas[i]) {
24             qtde++;
25             valor -= moedas[i];
26         }
27         printf("\n%d moeda(s) de R$ %.2f", qtde, moedas[i]);
28     }
29
30     return 0;
31 }
```

# Tentativa 7 (Python)



## SUBMISSÃO # 29461299

|            |                                 |
|------------|---------------------------------|
| PROBLEMA:  | 1021 - Notas e Moedas           |
| RESPOSTA:  | Accepted                        |
| LINGUAGEM: | Python 3.8 (Python 3.8.2) [+1s] |
| TEMPO:     | 0.084s                          |
| TAMANHO:   | 442 Bytes                       |
| MEMÓRIA:   | -                               |
| SUBMISSÃO: | 25/08/2022 18:45:33             |

## CÓDIGO FONTE

```
1  cedulas = [100, 50, 20, 10, 5, 2]
2  moedas = [100, 50, 25, 10, 5, 1]
3
4  valor = float(input())
5
6  print("NOTAS:")
7  for c in cedulas:
8      qtde = 0
9      while valor >= c:
10         qtde += 1
11         valor -= c
12         print("%d nota(s) de R$ %.2f" %(qtde, c))
13
14  print("MOEDAS:")
15  valor = int(valor * 100)
16  for m in moedas:
17      qtde = 0
18      while valor >= m:
19         qtde += 1
20         valor -= m
21         print("%d moeda(s) de R$ %.2f" %(qtde, m/100))
```

# Tentativa 8 (Python)



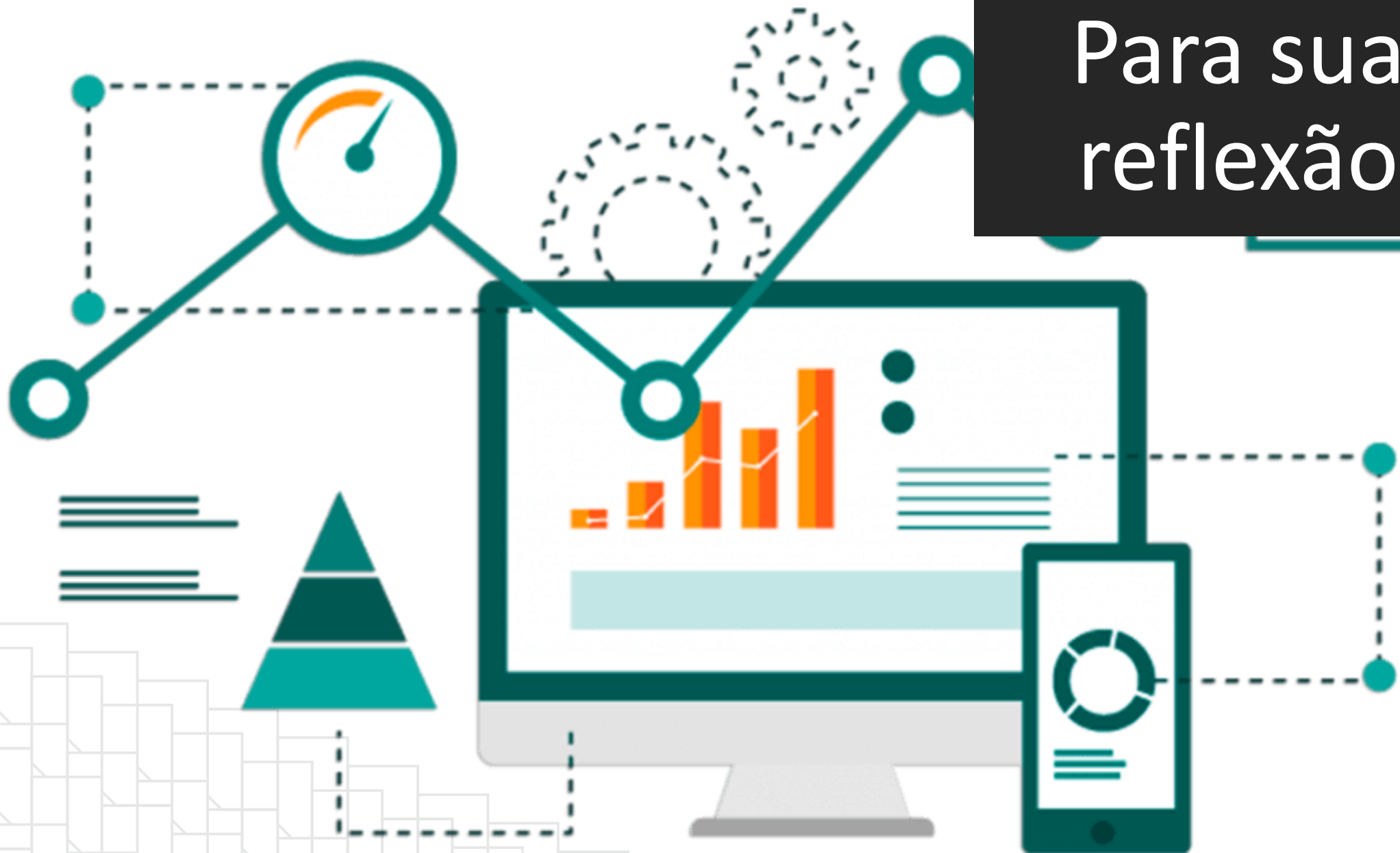
## SUBMISSÃO # 29827888

PROBLEMA: 1021 - Notas e Moedas  
RESPOSTA: **Accepted**  
LINGUAGEM: Python 3.8 (Python 3.8.2) [+1s]  
TEMPO: 0.070s  
TAMANHO: 461 Bytes  
MEMÓRIA: -  
SUBMISSÃO: 14/09/2022 17:14:39

## CÓDIGO FONTE

```
1  # -*- coding: utf-8 -*-
2
3  cedulas = [100, 50, 20, 10, 5, 2]
4  moedas = [100, 50, 25, 10, 5, 1]
5
6  valor = float(input())
7
8  print("NOTAS:")
9  for c in cedulas:
10     qtde = valor // c
11     valor = valor - (qtde*c)
12     print("%d nota(s) de R$ %.2f" %(qtde, c))
13
14  valor = valor * 100
15  # print(valor)
16  print("MOEDAS:")
17  for m in moedas:
18     qtde = valor // m
19     valor = valor - (qtde*m)
20     # print(valor, m)
21     print("%d moeda(s) de R$ %.2f" %(qtde, m/100))
```

Para sua  
reflexão





# Padawan – Exercício # 2 – Problema 1963

beecrowd | 1963

## O Filme

Por Leandro Zatesko, UFFS  Brazil

Timelimit: 1

Bem-vindo à 3ª Maratona de Programação *Júnior* da UFFS. Esperamos sinceramente que as próximas horas sejam muito produtivas para você, que você faça muitos balões e que, acima de tudo, você se divirta! Lembre que você sempre pode pedir esclarecimento quando não entender direito a descrição de um problema. Lembre também que às 17:30 os juízes automáticos serão desligados e a competição entrará em modo *blind*, de modo que todas as submissões neste período só começarão a ser julgadas às 18:10. Por favor, fique conosco até o fim da competição, trabalhando nas soluções dos problemas até o último minuto, pois, enquanto a competição ainda está ocorrendo, sempre há esperança!

E foi esperança que motivou a Vovó Zazá, uma senhora de 72 anos, a realizar seu sonho de começar um curso universitário. Ela está fascinada com tudo o que diz respeito à Universidade: com as aulas, com a biblioteca, com os projetos de pesquisa e extensão, com o restaurante universitário, mas especialmente com a carteirinha de estudante que ela pode utilizar para pagar meia entrada no cinema. Semana passada, Vovó Zazá e seus colegas de turma foram ao cinema assistir a um filme, mas ficaram estarecidos com o aumento do preço do ingresso. Revoltados, eles decidiram fazer uma manifestação contra o sistema capitalista opressor, agendada para amanhã na Praça General Bertaso. Vovó Zazá quer colaborar com o movimento fazendo um cartaz com a seguinte palavra de ordem:

QUE ABSURDO! O PREÇO DO CINEMA SUBIU ... % !!

Mas ela não é muito boa em Matemática, e está solicitando sua ajuda para calcular a porcentagem de que precisa para completar o cartaz.



# Padawan – Exercício # 2 – Problema 1963

## Entrada

A única linha da entrada consiste de dois valores **A** e **B** ( $0.00 < A \leq B \leq 1000.00$ ), os quais, fornecidos com exatos dois dígitos após o ponto separador decimal, representam respectivamente o valor antigo e o valor novo do ingresso do cinema.

## Saída

A única linha da saída deve consistir unicamente de um valor, que represente como uma porcentagem o aumento do valor do ingresso. O valor deve ser acompanhado do símbolo **%** e conter exatos dois dígitos após o ponto separador decimal.

| Exemplos de Entrada | Exemplos de Saída |
|---------------------|-------------------|
| 20.00 30.00         | 50.00%            |
| 50.00 100.00        | 100.00%           |
| 10.00 10.00         | 0.00%             |



# Resolução 1

## C

### Exemplos de Saída

50.00%

100.00%

0.00%

SUBMISSÃO 32174178

LINGUAGEM

C99

RESPOSTA

Accepted

RUNTIME

0.018s

### CÓDIGO FONTE

```
1  #include <stdio.h>
2
3  int main()
4  {
5
6      float a, b, r;
7      scanf("%f", &a);
8      scanf("%f", &b);
9
10     // (oldPrice - newPrice) / newPrice
11     r = ((b - a) / a) * 100;
12
13     printf("%.2f%%\n", r);
14
15     return 0;
16 }
```



# Resolução 2

## Python

### Exemplos de Saída

50.00%

100.00%

0.00%

SUBMISSÃO 32169855

LINGUAGEM

PYTHON 3.8

RESPOSTA

Accepted

RUNTIME

0.119s

### CÓDIGO FONTE


```
1 old, new = input().split()
2
3 old = float(old)
4 new = float(new)
5 per = ((new*100)/old)-100
6
7 print("%.2f%%" %(per))
```



# Cavaleiro Jedi – Exercício # 1 – Problema 1045

beecrowd | 1045

## Tipos de Triângulos

Adaptado por Neilor Tonin, URI  Brasil

Timelimit: 1

---

Leia 3 valores de ponto flutuante A, B e C e ordene-os em ordem decrescente, de modo que o lado A representa o maior dos 3 lados. A seguir, determine o tipo de triângulo que estes três lados formam, com base nos seguintes casos, sempre escrevendo uma mensagem adequada:

- se  $A \geq B+C$ , apresente a mensagem: **NAO FORMA TRIANGULO**
- se  $A^2 = B^2 + C^2$ , apresente a mensagem: **TRIANGULO RETANGULO**
- se  $A^2 > B^2 + C^2$ , apresente a mensagem: **TRIANGULO OBTUSANGULO**
- se  $A^2 < B^2 + C^2$ , apresente a mensagem: **TRIANGULO ACUTANGULO**
- se os três lados forem iguais, apresente a mensagem: **TRIANGULO EQUILATERO**
- se apenas dois dos lados forem iguais, apresente a mensagem: **TRIANGULO ISOSCELES**

### Entrada

A entrada contém três valores de ponto flutuante de dupla precisão A ( $0 < A$ ), B ( $0 < B$ ) e C ( $0 < C$ ).

### Saída

Imprima todas as classificações do triângulo especificado na entrada.



# Cavaleiro Jedi – Exercício # 1 – Problema 1045

| Exemplos de Entrada | Exemplos de Saída                            |
|---------------------|--|
| 7.0 5.0 7.0         | TRIANGULO ACUTANGULO<br>TRIANGULO ISOSCELES  |
| 6.0 6.0 10.0        | TRIANGULO OBTUSANGULO<br>TRIANGULO ISOSCELES |
| 6.0 6.0 6.0         | TRIANGULO ACUTANGULO<br>TRIANGULO EQUILATERO |
| 5.0 7.0 2.0         | NAO FORMA TRIANGULO                          |
| 6.0 8.0 10.0        | TRIANGULO RETANGULO                          |





# Resolução 1

## Python

### Exemplos de Saída

```
TRIANGULO ACUTANGULO
TRIANGULO ISOSCELES
```

```
TRIANGULO OBTUSANGULO
TRIANGULO ISOSCELES
```

```
TRIANGULO ACUTANGULO
TRIANGULO EQUILATERO
```

```
NAO FORMA TRIANGULO
```

```
TRIANGULO RETANGULO
```

SUBMISSÃO 32195151

LINGUAGEM

PYTHON 3

RESPOSTA

Accepted

RUNTIME

0.164s

### CÓDIGO FONTE

```
1  A, B, C = input().split()
2  A = float(A)
3  B = float(B)
4  C = float(C)
5  A, B, C = sorted([A, B, C], reverse = True)
6  if A >= (B + C):
7      print("NAO FORMA TRIANGULO")
8  if (A ** 2) == (B ** 2 + C ** 2):
9      print("TRIANGULO RETANGULO")
10 if (A ** 2) > (B ** 2 + C ** 2) and not(A >= (B + C)):
11     print("TRIANGULO OBTUSANGULO")
12 if (A ** 2) < (B ** 2 + C ** 2):
13     print("TRIANGULO ACUTANGULO")
14 if A == B and A == C:
15     print("TRIANGULO EQUILATERO")
16 if (A == B and A != C) or (A == C and A != B) or (B == C and B != A):
17     print("TRIANGULO ISOSCELES")
```



# Resolução 2

## C++

### Exemplos de Saída

TRIANGULO ACUTANGULO  
TRIANGULO ISOSCELES

TRIANGULO OBTUSANGULO  
TRIANGULO ISOSCELES

TRIANGULO ACUTANGULO  
TRIANGULO EQUILATERO

NAO FORMA TRIANGULO

TRIANGULO RETANGULO

SUBMISSÃO 32181546

LINGUAGEM

C++17

RESPOSTA

Accepted

RUNTIME

0.000s

### CÓDIGO FONTE

```

1  #include <iostream>
2  #include <string>
3  #include <vector>
4  #include <sstream>
5  #include <math.h>
6
7  int main() {
8      std::string str = "";
9      std::getline(std::cin, str);
10
11     std::stringstream Stringstream(str);
12     std::vector<double> lados;
13     double value = 0.0;
14
15     while(Stringstream >> value){
16         lados.push_back(value);
17     }
18
19     if(lados[2] >= (lados[1] + lados[0]) || lados[1] >= (lados[2] + lados[0]) || lados[0] >= (lados[1]
+lados[2]))
20         std::cout << "NAO FORMA TRIANGULO\n";
21     else if(pow(lados[2],2) == pow(lados[1],2) + pow(lados[0],2) || pow(lados[1],2) == pow(lados[2],2) + pow
(lados[0],2) || pow(lados[0],2) == pow(lados[1],2) + pow(lados[2],2))
22         std::cout << "TRIANGULO RETANGULO\n";
23     else if(pow(lados[2],2) > pow(lados[1],2) + pow(lados[0],2) || pow(lados[1],2) > pow(lados[2],2) + pow
(lados[0],2) || pow(lados[0],2) > pow(lados[1],2) + pow(lados[2],2))
24         std::cout << "TRIANGULO OBTUSANGULO\n";
25     else if(pow(lados[2],2) < pow(lados[1],2) + pow(lados[0],2) || pow(lados[1],2) < pow(lados[2],2) + pow
(lados[0],2) || pow(lados[0],2) < pow(lados[1],2) + pow(lados[2],2))
26         std::cout << "TRIANGULO ACUTANGULO\n";
27     if(lados[2] == lados[1] && lados[2] == lados[0] && lados[1] == lados[0])
28         std::cout << "TRIANGULO EQUILATERO\n";
29     if(((lados[2] == lados[1]) && (lados[2] != lados[0])) || ((lados[2] == lados[0]) && (lados[2] !=
lados[1])) || ((lados[1] == lados[0]) && (lados[2] != lados[1]))
30         std::cout << "TRIANGULO ISOSCELES\n";
31     return 0;
32 }
```





# Cavaleiro Jedi – Exercício # 2 – Problema 1984

beecrowd | 1984



## O Enigma do Pronalândia

Por Viktor Jatobá, UNIME  Brazil

Timelimit: 1

Os cientistas da NASA descobriram um novo exoplaneta que fica a 1 bilhão de anos luz da terra. O nome desse planeta foi batizado de Pronalândia em homenagem aos novos cientistas que estão sendo formados no PRONATEC. Só que o mais incrível ainda está por vir. Ao observar melhor o planeta eles conseguiram identificar que os habitantes da Pronalândia estavam querendo se comunicar por uma numeração. Só que a numeração que encontraram está invertida e como encontraram muitas delas chamaram você para conseguir automatizar esse processo. Logo, dado um número grande, sua tarefa é imprimir esse número invertido.

### Entrada

O arquivo contém apenas uma linha de teste que é o número encontrado ( $0 < n < 9999999999$ ).

Obs.: Perceba que o número lido é muito alto para armazenar em uma variável do tipo int, logo você irá precisar utilizar o tipo long, que para a leitura e impressão em C, você deve utilizar o %llu.

### Saída

Imprimir o número lido invertido. Não esqueça de imprimir a quebra de linha ( $\backslash n$ ) no final, caso contrário você receberá (Presentation Error).



# Cavaleiro Jedi – Exercício # 2 – Problema 1984

| Exemplo de Entrada | Exemplo de Saída |
|--------------------|------------------|
| 1234               | 4321             |
| 9876543210         | 0123456789       |
| 12                 | 21               |

I Seletiva MaratonaTEC - Maratona de Programação PRONATEC - UNIME



# Resolução 1

## C++

### Exemplo de Saída

4321

0123456789

21



SUBMISSÃO 32176441

LINGUAGEM

C++17

RESPOSTA

Accepted

RUNTIME

0.000s

### CÓDIGO FONTE

```
1  #include <iostream>
2  #include <string>
3  #include <algorithm>
4
5  using namespace std;
6
7  int main() {
8      long long int enigma;
9      string numero;
10
11     cin >> enigma;
12     numero = to_string(enigma);
13     reverse(numero.begin(), numero.end());
14
15     for(int i = 0; i < numero.size(); ++i)
16         cout << numero[i];
17     cout << "\n";
18
19     return 0;
20 }
```

# Resolução 2

## Python

### Exemplo de Saída

4321

0123456789

21

## SUBMISSÃO 32186401

LINGUAGEM

PYTHON 3.9

RESPOSTA

Accepted

RUNTIME

0.000s

### CÓDIGO FONTE

```
1 print(str(int(input()))[::-1])
```



# Mestre Yoda – Exercício # 1 – Problema 1973

beecrowd | 1973

**Jornada nas Estrelas**

Por Leandro Zatesko, UFFS  Brazil

Timelimit: 1

---

Após comprar vários sítios adjacentes na região do oeste catarinense, a família Estrela construiu uma única estrada que passa por todos os sítios em sequência. O primeiro sítio da sequência foi batizado de Estrela 1, o segundo de Estrela 2, e assim por diante. Porém, o irmão que vive em Estrela 1 acabou enlouquecendo e resolveu fazer uma Jornada nas Estrelas para roubar carneiros das propriedades de seus irmãos. Mas ele está definitivamente pirado. Quando passa pelo sítio Estrela  $i$ , ele rouba apenas um carneiro daquele sítio (se o sítio tem algum) e segue ou para Estrela  $i + 1$  ou para Estrela  $i - 1$ , dependendo se o número de carneiros em Estrela  $i$  era, respectivamente, ímpar ou par. Se não existe a Estrela para a qual ele deseja seguir, ele interrompe sua jornada. O irmão louco começa sua Jornada em Estrela 1, roubando um carneiro do seu próprio sítio.

### Entrada

A primeira linha da entrada consiste de um único inteiro  $N$  ( $1 \leq N \leq 10^6$ ), o qual representa o número de Estrelas. A segunda linha da entrada consiste de  $N$  inteiros, de modo que o  $i$ -ésimo inteiro,  $X_i$  ( $1 \leq X_i \leq 10^6$ ), representa o número inicial de carneiros em Estrela  $i$ .

### Saída

Imprima uma linha contendo dois inteiros, de modo que o primeiro represente o número de Estrelas atacadas pelo irmão louco e o segundo represente o número total de carneiros *não* roubados.



# Mestre Yoda – Exercício # 1 – Problema 1973

## 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



# Resolução 1

## Python

SUBMISSÃO 32200532

LINGUAGEM

PYTHON 3.8

RESPOSTA

Accepted

RUNTIME

0.510s

Exemplo de Saída

8 68

7 63

### CÓDIGO FONTE

```
1 def main():
2     input();r=list(map(int,input().split()));v=s=o=0
3     for x in r:
4         if x%2==1:v+=1;s+=x-1;o+=x==1
5         else:s+=x-1-v+o+sum(r[v+1:]);v+=1;break
6     print(v,s)
7     main()
```



# Resolução 2

## Python

SUBMISSÃO 32208423

LINGUAGEM

PYTHON 3.9

RESPOSTA

Accepted

RUNTIME

0.994s

### Exemplo de Saída

8 68

7 63

### CÓDIGO FONTE

```
1 import sys
2 from collections import deque
3 from itertools import islice
4
5 def main() -> str:
6     shape = int(input())
7     content = deque(map(int, sys.stdin.readline().split()))
8
9     is_even = deque(filter(lambda item: item % 2 == 0, content))
10
11     if is_even:
12         first_even = content.index(is_even[0])+1
13         sheep_before_first = len(list(filter(lambda item: item == 1, islice(content, 0, first_even))))
14         return f'{first_even} {sum(content) - (((first_even-1)*2)+1) - sheep_before_first}'
15
16     else:
17         return f'{shape} {sum(content) - shape}'
18
19 print(main())
```





# Mestre Yoda – Exercício # 1 – Problema 2163

beecrowd | 2163

O Despertar da Força

Por M.C. Pinto, UNILA  Brazil

Timelimit: 1

---

Há muito tempo atrás, em uma galáxia muito, muito distante...

Após o declínio do Império, sucateiros estão espalhados por todo o universo procurando por um sabre de luz perdido. Todos sabem que um sabre de luz emite um padrão de ondas específico: 42 cercado por 7 em toda a volta. Você tem um sensor de ondas que varre um terreno com  $N \times M$  células. Veja o exemplo abaixo para um terreno  $4 \times 7$  com um sabre de luz nele (na posição (2, 4)).

|    |    |   |    |   |    |    |
|----|----|---|----|---|----|----|
| 11 | 12 | 7 | 7  | 7 | 13 | 14 |
| 15 | 6  | 7 | 42 | 7 | 7  | 42 |
| 98 | -5 | 7 | 7  | 7 | 42 | 7  |
| -1 | 42 | 3 | 9  | 7 | 7  | 7  |

Você deve escrever um programa que, dado um terreno  $N \times M$ , procura pelo padrão do sabre de luz nele. Nenhuma varredura tem mais do que um padrão de sabre de luz.

# Mestre Yoda – Exercício # 1 – Problema 2163

## Entrada

A primeira linha da entrada tem dois números positivos **N** e **M**, representando, respectivamente, o número de linhas e de colunas varridos no terreno ( $3 \leq N, M \leq 1000$ ). Cada uma das próximas **N** linhas tem **M** inteiros, que descrevem os valores lidos em cada célula do terreno ( $-100 \leq T_{ij} \leq 100$ , para  $1 \leq i \leq N$  e  $1 \leq j \leq M$ ).

## Saída

A saída é uma única linha com 2 inteiros **X** e **Y** separados por um espaço. Eles representam a coordenada (**X,Y**) do sabre de luz, caso encontrado. Se o terreno não tem um padrão de sabre de luz, **X** e **Y** são ambos zero.

| Exemplos de Entrada  | Exemplos de Saída |
|--|-------------------|
| <pre>4 7 11 12 7 7 7 13 14 15 6 7 42 7 7 42 98 -5 7 7 7 42 7 -1 42 3 9 7 7 7</pre> | <pre>2 4</pre>    |
| <pre>4 7 11 12 7 7 7 13 14 15 6 7 12 7 7 42 98 -5 7 7 7 42 7 -1 42 3 9 7 7 7</pre> | <pre>0 0</pre>    |
| <pre>3 3 7 7 7 7 42 7 7 7 7</pre>  | <pre>2 2</pre>    |



# Resolução 1

## Python

SUBMISSÃO 32208759

LINGUAGEM

PYTHON 3.9

RESPOSTA

Accepted

RUNTIME

1.347s

### Exemplos de Saída

2 4

0 0

2 2

### CÓDIGO FONTE

```
1 def main() -> str:
2     row = input().split()
3     shape_y = int(row[0])
4     shape_x = int(row[1])
5
6     content = []
7
8     for i in range(shape_y):
9         current_input = input().split()
10
11         current_row = [int(x) for x in current_input]
12
13         content.append(current_row)
14
15     for y in range(1, shape_y-1):
16         for x in range(1, shape_x-1):
17             if content[y][x] == 42:
18                 total = content[y-1][x-1] + content[y-1][x] + content[y-1][x+1] + content[y+1][x-1] +
content[y+1][x] + content[y+1][x+1] + content[y][x-1] + content[y][x+1]
19                 if total == 56:
20                     return f'{y+1} {x+1}'
21
22     return '0 0'
23
24 print(main())
```



# Resolução 2

## C++

### Exemplos de Saída

2 4

0 0

2 2



SUBMISSÃO 32169022

LINGUAGEM

C++17

RESPOSTA

Accepted

RUNTIME

0.599s

### CÓDIGO FONTE

```

1  #include <bits/stdc++.h>
2  #define MAXN 1007
3
4  using namespace std;
5
6  int main(){
7      int n, m, matriz[MAXN][MAXN];
8      cin >> n >> m;
9
10     memset(matriz, -1, sizeof(matriz));
11
12     for(int i = 1; i <= n; i++){
13         for(int j = 1; j <= m; j++){
14             cin >> matriz[i][j];
15         }
16     }
17
18     bool x = false;
19
20     for(int i = 1; i <= n; i++){
21         for(int j = 1; j <= m; j++){
22             if(matriz[i][j] == 42 && matriz[i-1][j] == 7 && matriz[i+1][j] == 7 && matriz[i][j-1] == 7 &&
matriz[i][j+1] == 7 && matriz[i+1][j+1] == 7 && matriz[i+1][j-1] == 7 && matriz[i-1][j-1] == 7 && matriz[i-1][j+1] == 7){
23                 cout << i << " " << j << endl;
24                 x = true;
25             }
26         }
27     }
28
29     if(!x){
30         cout << 0 << " " << 0 << endl;
31     }
32
33     return 0;
34
35 }
```

Responder o formulário a seguir para contabilizar a sua presença:

- <https://forms.office.com/r/BvUmw7s6cT>



The image shows a screenshot of a Google Form titled "Quiz #2 - 2023/1 - Treinamento Maratona de Programação". The form is overlaid on a background image of a modern building with many windows. The text on the form is as follows:

**Quiz #2 - 2023/1 - Treinamento Maratona de Programação**

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

Olá, IVAN. Quando você enviar este formulário, o proprietário verá seu nome e endereço de email.

\* Obrigatória

1. Nos problemas que você tentou resolver no Beecrowd, selecione as mensagens recebidas nas suas soluções \*

The background of the slide is a dark, textured surface covered with numerous 3D question marks. Most of these question marks are dark grey or black, but one in the center-left is a bright, glowing white. The lighting creates soft shadows and highlights on the question marks, giving them a three-dimensional appearance.

Dúvidas!?



# Hands On



