



Roadmap Java



Módulo 1

Fundamentos

- **Lógica de programação e algoritmos**
- **Sintaxe da linguagem Java**
- **Classes, atributos e métodos**
- **Construtores, sobrecarga, encapsulamento**
- **Estruturas de dados: arrays e coleções**
- **Enumerações, composição**
- **Herança, polimorfismo, interfaces**
- **Tratamento de exceções**



Material de Estudo

Fundamentos

- **Nesse módulo iremos estudar o fundamentos da classe e após isso executar alguns desafios propostos, a seguir no material de apoio temos 2 cursos que para apoiá-los nos desafios**

Material de Apoio

- [Curso Online Java: Caelum Java e Orientação a Objetos - Como Aprender Java](#)
- [Curso Online Java: Caelum Java e Orientação a Objetos - O que é Java](#)
- [Curso Online Java: Caelum Java e Orientação a Objetos - Variáveis Primitivas e Controle de Fluxo](#)
- [Curso Online Java: Caelum Java e Orientação a Objetos - Orientação a Objetos Básica](#)
- [Curso Online Java: Udemmy - Java Completo - Seção 1 a 15](#)



Desafios

Considerações

- Para a execução desses desafios, criar um projeto no git com o nome Modulo_1
- O prazo de Execução é de 10 dias:
- Início: 13/10
- Fim: 22/10

Boa Sorte!!!!

Desafio 1

Teleférico

A turma da faculdade vai fazer uma excursão na serra e todos os alunos e monitores vão tomar um teleférico para subir até o pico de uma montanha. A cabine do teleférico pode levar C pessoas no máximo, contando alunos e monitores, durante uma viagem até o pico. Por questão de segurança, deve haver pelo menos um monitor dentro da cabine junto com os alunos. Por exemplo, se cabem $C = 10$ pessoas na cabine e a turma tem $A = 20$ alunos, os alunos poderiam fazer três viagens:

- a primeira com 8 alunos e um monitor;
- a segunda com 6 alunos e um monitor;
- e a terceira com 6 alunos e um monitor.

Dados como entrada a capacidade C da cabine e o número total A de alunos, você deve escrever um programa para calcular o número mínimo de viagens do teleférico.

Desafio 1

Entrada

A primeira linha da entrada contém um inteiro C , representando a capacidade da cabine ($2 \leq C \leq 100$). A segunda linha da entrada contém um inteiro A , representando o número total de alunos na turma ($1 \leq A \leq 1000$).

Saída

Seu programa deve imprimir uma linha contendo um número inteiro representando o número mínimo de viagens do teleférico para levar todos os alunos até o pico da montanha.

Exemplo de entrada 1 10 20	Exemplo de saída 1 3
Exemplo de entrada 2 12 55	Exemplo de saída 2 5
Exemplo de entrada 3 100 87	Exemplo de saída 3 1

Desafio 2

Fatorial

O fatorial de um número inteiro positivo N , denotado por $N!$, é definido como o produto dos inteiros positivos menores do que ou iguais a N . Por exemplo $4! = 4 \times 3 \times 2 \times 1 = 24$. Dado um inteiro positivo N , você deve escrever um programa para determinar o menor número k tal que $N = a_1! + a_2! + \dots + a_k!$, onde cada a_i , para $1 \leq i \leq k$, é um número inteiro positivo. Por exemplo, para $N = 10$ a resposta é 3, pois é possível escrever N como a soma de três números fatoriais: $10 = 3! + 2! + 2!$. Para $N = 25$ a resposta é 2, pois é possível escrever N como a soma de dois números fatoriais: $25 = 4! + 1!$.

Desafio 2

Entrada

A entrada consiste de uma única linha que contém um inteiro N ($1 \leq N \leq 10^5$).

Saída

Seu programa deve produzir uma única linha com um inteiro representando a menor quantidade de números fatoriais cuja soma é igual ao valor de N .

Exemplo de entrada 1 10	Exemplo de saída 1 3
Exemplo de entrada 2 25	Exemplo de saída 2 2

Desafio 3

Nota Esquecida

João não é muito bom em Matemática mas aprendeu que a média de dois números é o valor da soma desses dois números dividido por dois. Ou seja, a média de dois números A e B é $M = \frac{A+B}{2}$. A professora de Cálculo I contou para João as notas que ele tirou nas duas provas já realizadas. As duas notas são números inteiros entre 0 e 100. João prontamente calculou a média das duas provas, que também resultou em um número inteiro. Mas João é muito esquecido, e agora não consegue lembrar-se das duas notas que tirou na prova. Ele consegue se lembrar de apenas uma das notas das provas. Por sorte, ele consegue se lembrar também da média entre as duas notas. Você pode ajudar João a determinar a nota da outra prova?

Desafio 3

Entrada

A primeira linha contém um número inteiro A ($0 \leq A \leq 100$), indicando a nota de uma prova. A segunda linha contém um número inteiro M ($0 \leq M \leq 100$), indicando a média entre as duas notas das provas.

Saída

Seu programa deve produzir uma única linha com um número inteiro representando a nota da outra prova, que João não consegue recordar.

Exemplo de entrada 1 100 70	Exemplo de saída 1 40
Exemplo de entrada 2 80 75	Exemplo de saída 2 70
Exemplo de entrada 3 1 50	Exemplo de saída 3 99

Cigarras periódicas

As “cigarras periódicas” americanas tem o ciclo de vida mais longo de todos os insetos conhecidos. A cada 17 anos, estas cigarras periódicas amadurecem, se acasalam, depositam ovos e morrem. Suas crias se refugiam debaixo da terra, a 20 centímetros de profundidade, onde elas se alimentam da seiva de raízes por 17 anos, até que chegue seu dia de buscar um lugar ao sol. Acredita-se que esse número não aconteceu por acaso, outras espécies de cigarras da região têm ciclos de 13 anos, assim essas duas espécies emergem ao mesmo tempo apenas a cada 221 anos. Isso é desejável pois dessa forma a chance de que as duas espécies se misturem diminui consideravelmente e características indesejáveis de uma população não são introduzidas na outra. Inspirado por esse fenômeno, uma nova variação de algoritmo evolutivo foi criada. Na última etapa desse algoritmo as melhores possíveis soluções são divididas em populações de modo que cada população I tem um ciclo de vida C_i . Além disso uma população extra também é adicionada, de modo que a quantidade de iterações até que o ciclo de vida de todas as populações coincida seja a maior possível. Essas populações são então avaliadas até que o ciclo de vida de todas coincida é a melhor solução ao final do processo é escolhida. Como não é interessante esperar demais até que o algoritmo gere uma resposta, um limite superior L no número de iterações também deve ser respeitado. Dados os ciclos de vida das populações criadas e o limite na quantidade de iterações L , sua tarefa é computar qual o período ótimo para a população extra que será adicionada.

Desafio Plus

Entrada

A primeira linha da entrada contém dois inteiros N e L , respectivamente, a quantidade de populações geradas pelas etapas anteriores do algoritmo e o limite da quantidade de iterações, $2 \leq N \leq 10^4$, $1 \leq L \leq 10^6$. A linha seguinte contém os N valores C_i representando a quantidade de iterações no ciclo de vida de cada população, onde $1 \leq C_i$. Você pode assumir que os ciclos de vida das populações atuais coincidem em menos de L iterações.

Saída

Seu programa deve produzir uma única linha com um inteiro representando o período da população extra que maximiza a quantidade T de iterações até que os ciclos de vida de todas as populações coincidam, respeitando a restrição de que $T \leq L$. Caso exista mais de um valor possível imprima o menor deles.

Exemplo de entrada 1 2 5000 105 55	Exemplo de saída 1 4
Exemplo de entrada 2 2 512 3 14	Exemplo de saída 2 72
Exemplo de entrada 3 3 80 6 10 15	Exemplo de saída 3 4



Obrigado