**Reajuste de salário**

João é um empresário do ramo de construção civil e decidiu adotar novos critérios para dar reajuste salarial para seus funcionários. Agora, o reajuste vai ser dado seguindo as regras abaixo.

Tempo de serviço:

* de 1 até 5 anos => 1%
* de 5 até 10 anos => 1.5%
* 10 ou mais anos => 2%

O valor total percentual para o reajuste vai ser a soma do percentual de tempo de serviço mais o percentual da inflação.

Para ajudar João a calcular o valor do reajuste salarial de seus funcionários, crie uma função que recebe um array como entrada. A posição 0 do array é o tempo de serviço, a posição 1 é o valor da inflação e a posição 3 é o salário atual do funcionário. Sua função deve calcular o novo salário e retorná-lo.

Considere arredondamento de duas casas decimais para o salário retornado.

Ex:

Entrada: [1, 5.0, 2000.00]

novoSalario = 2000.00 \* (1 + (1% + 5%)) = 2000.00 \* 1.06

Saída: 2120.00

Ex:

Entrada: [11, 4.5, 2500.00]

novoSalario = 2500.00 \* (1 + (2% + 4.5%)) = 2500.00 \* 1.065

Saída: 2662.50

Resultados

Inicie com esse código

public class Solution {

    public static double calculoSalario( double[] input) {

        return 0.0;

    }

}

# Compra do carro novo

Maria está olhando o mercado de automóveis para comprar um carro novo. Ela identificou que o preço final do veículo tem um percentual relacionado ao distribuidor e também um percentual de impostos. Ela deseja identificar qual veículo possui os menores percentuais de imposto e do distribuidor, para fazer a escolha do seu carro novo.

Para ajudar Maria em sua busca, crie uma função que receba um array com o preço de final de fábrica, o valor do custo com o distribuidor e o valor do preço dos impostos. Ao final, a função deve retornar um array com o percentual do custo do distribuidor e o percentual do custo com os impostos, seguindo essa ordem.

Realize o arredondamento para duas casas decimais em relação aos dados de retorno.

Ex:

Entrada: [100000.00, 12000.00, 20000.00]

Saída: [12.00, 20.00]

Ex:

Entrada: [115500.00,25000.00, 32500.00]

Saída: [21.64, 28.14]

### Resultados

public class Solution {

    public static double[] custosCarro( double[] input) {

        return null;

    }

}

# Quitanda da Maria

Maria tem uma frutaria e está com uma promoção para quem comprar mais de 10 frutas (do mesmo item). As frutas com preços especiais são: Pera, Laranja, Maçã, Tomate e Caqui.

Comprando 10 unidades ou mais de qualquer fruta da promoção, o preço da unidade fica 1.25/cada. Caso a pessoa leve uma quantidade inferior ou igual a 10, o preço individual da fruta fica em 1.45.

Faça uma função que recebe a quantidade de frutas que o cliente está levando e calcule o valor final a ser pago.

Arredonde o valor final da compra para 2 casas decimais.

Ex:

Entrada: 2

Saída: 2.90

Ex:

Entrada: 12

Saída: 15.00

### Resultados

public class Solution {

    public static double custoCompra(double input) {

        return 0.0;

    }

}

# Arrays, Laços e Condicionais

Dentro dos conjuntos numéricos existem vários subconjuntos, dentre eles os subconjuntos dos números pares e dos números ímpares. Os números pares são todos os números múltiplos de 2, enquanto os números ímpares são o números não pares, ou seja, são os números que não são múltiplos de 2. Esse conjunto de números terminam com algum dos algarismos 1, 3, 5, 7 ou 9.

Crie uma função que recebe um array de números inteiros, identifica os números como pares ou ímpares e retorna uma String informando a quantidade de pares, seguido de uma vírgula e, depois, a quantidade de ímpares.

Obs.: Atenção para a flexão do plural no texto de saída

Ex:

Entrada: 1,2,3,6,9

Saída: 3 pares, 3 ímpares

Ex:

Entrada: 2,3,6

Saída: 2 pares, 1 ímpar

### Resultados

public class Solution {

    public static String parImpar(int[] input) {

        return "";

    }

}

# Arrays, Ordenação, Operações aritméticas

Uma loja de roupas decidiu aquecer suas vendas lançando uma promoção com descontos especiais para clientes que comprassem mais de uma peça. As condições da promoção dizem que, na compra de duas peças, será dado 50% de desconto na peça de menor de valor.

Escreva uma função que receba um array contendo o preço das peças de um cliente e calcule o valor total sem descontos, o valor total de descontos, e o valor total a pagar. A saída da função deve ser uma String.

Ex:

Entrada: [20.0, 40.0]

Saída: Valor total: 60.0 | Valor de descontos: 10.0 | Valor a pagar: 50.0

Ex:

Entrada: [20.0, 40.0, 30.0]

Saída: Valor total: 90.0 | Valor de descontos: 10.0 | Valor a pagar: 80.0

Ex:

Entrada: [20.0, 80.0, 60.0, 40.0]

Saída: Valor total: 200.0 | Valor de descontos: 10.0 | Valor a pagar: 190.0

### Resultados

public class Solution {

    public static String geraRecibo(double[] input) {

        return "";

    }

}

**O vendedor de carros**

Pedrinho trabalha com vendas de carro. Mensalmente, ele recebe um salário fixo mais comissões baseadas em suas vendas. As comissões são as seguintes:

* 5% sobre o valor total vendido no mês
* valor fixo por cada carro vendido

Faça uma função que recebe um array com: quantidade de carros vendidos por ele no mês, o valor total de suas vendas no mês, seu salário fixo, valor fixo que recebido por carro vendido. Calcule e retorne qual o salário dele (salário fixo mais comissões).

Arredonde o valor de retorno para duas casas decimais.

Ex:

Entrada: [3, 20000.00, 2000.00, 250.00]

Saída: 3750.00

Ex:

Entrada: [4,25000.00, 3500.00, 100.00]

Saída: 5150.00

Resultados

public class Solution {

    public static double salarioComComissao( double[] input) {

        return 0.0;

    }

}

**Condicionais e Conversão**

Em uma danceteria o preço da entrada sofre variações. Segundas, Terças e Quintas, ela oferece um desconto de 25% do preço normal de entrada. Nos dias de músicas ao vivo, o preço da entrada ainda é acrescido em 15% em relação ao preço normal da entrada.

Faça uma função que recebe o dia da semana, o preço normal da entrada e se é dia de música ao vivo (“Sim”) ou não (“Não”). Os valores de entrada serão passados em um array seguindo essa ordem. Calcule e retorne o preço final que deverá ser pago pela entrada. Considere um arredondamento de três casas para o valor final da entrada.

Obs.: aplicar primeiro o desconto dos dias com desconto e depois o acréscimo de música ao vivo, quando os dois descontos forem aplicados.

A entrada dos dias da semana será considerando um valor decimal, conforme os dados a seguir:

* 1.0 - segunda
* 2.0 - terça-feira
* 3.0 - quarta-feira
* 4.0 - quinta-feira
* 5.0 - sexta-feira
* 6.0 - sábado
* 7.0 - domingo

Para os dados referentes a música ao vivo, serão considerados: 1.0 - sim e 2.0 - não.

Ex:

Entrada: [3.0, 25.00, 2.0]

Saída: 25.000

Ex:

Entrada: [4.0, 50.00, 1.0]

Saída: 43.125

Resultados

public class Solution {

    public static double valorEntrada(double[] input) {

        return 0.0;

    }

}