INSTITUTO FEDERAL
Goiano
Campus Morrinhos

Programação Orientada a Objetos
Curso Tecnologia em Sistemas para Internet
Professor Msc. Marcel Melo

---

1. Se você precisa construir um componente usando o Raspberry Pi, provavelmente usará resistores. Para este exercício, você precisa saber duas coisas sobre eles:
   - Cada resistor tem um valor de resistência.
   - Os resistores são pequenos - tão pequenos que, se você imprimir o valor da resistência neles, seria difícil de ler.

   Para resolver esse problema, os fabricantes imprimem faixas codificadas por cores nos resistores para definir seus valores de resistência. Cada faixa tem uma posição e um valor numérico.
   As 2 primeiras faixas de um resistor têm um esquema de codificação simples: cada cor é mapeada para um número único. Por exemplo, se eles imprimissem uma faixa marrom (valor 1) seguida por uma faixa verde (valor 5), seria traduzido para o número 15.

   Neste exercício, você criará um programa em Java para não precisar se lembrar dos valores das faixas. O programa receberá nomes de cores como entrada e produzirá um número de dois dígitos, mesmo que a entrada tenha mais de duas cores! As cores da faixas são codificadas da seguinte forma:

   - Preto: 0
   - Marrom: 1
   - Vermelho 2
   - Laranja: 3
   - Amarelo: 4
   - Verde: 5
   - Azul: 6
   - Violeta: 7
   - Cinza: 8
   - Branco: 9

   Do exemplo acima: marrom-verde deve retornar 15 marrom-verde-violeta também deve retornar 15, ignorando a terceira cor.

2. Você precisa calcular o salário de um funcionário ao final do mês. No entanto, existem alguns fatores que podem incrementar ou decrementar o salário deste funcionário.
   - Se o funcionário faltou mais que 5 vezes no mês, ele deve ter o salário descontado em 15%;

- Para cada produto que o funcionário vendeu, você deve dar um bônus de R$10,00 por produto; No entanto se ele vendeu mais que 20 produtos no mês, o bônus é de $13,00 por produto vendido.

3. Você precisa desenvolver um sistema que permita o usuário digitar vários números inteiros, positivos ou negativos. Ao final, quando o usuário digita o valor -1, o programa deve calcular e exibir:
    - A quantidade de números digitados pelo usuário;
    - O maior número digitado;
    - O menor número digitado;
    - A média dos números digitados;

   Para esse exercício você NÃO deve trabalhar com vetores ou *ArrayList*.

4. Dado um número, determine se ele é ou não válido de acordo com a fórmula de Luhn. O algoritmo Luhn é uma fórmula de soma de verificação simples usada para validar uma variedade de números de identificação, como números de cartão de crédito. O objetivo deste exercício é determinar se uma String de números é ou não válida.

Validando um número

- Strings de comprimento 1 ou menos não são válidas. Espaços são permitidos na entrada, mas devem ser removidos antes da verificação. Todos os outros caracteres que não sejam dígitos não são permitidos.

Exemplo: Número de cartão de crédito válido

4539 3195 0343 6467

A primeira etapa do algoritmo de Luhn é remover os espaços e dobrar cada segundo dígito, começando pela direita. nós estaremos dobrando

4_3_ 3_9_ 0_4_ 6_6_

Se dobrar o número resultar em um número maior que 9, subtraia 9 do produto. Os resultados da nossa duplicação:

8569 6195 0383 3437

Em seguida, some todos os dígitos:

8+5+6+9+6+1+9+5+0+3+8+3+3+4+3+7 = 80

Se a soma for divisível por 10, o número é válido. Este número é válido! Caso contrário, o número é inválido.

5 - Calcule a distância de Hamming entre duas cadeias de DNA.

Seu corpo é feito de células que contêm DNA. Essas células se desgastam regularmente e precisam ser substituídas, o que elas conseguem ao se dividir em células-filhas. Na verdade, o corpo humano médio passa por cerca de 10 quatrilhões de divisões celulares durante a vida!

Quando as células se dividem, seu DNA também se replica. Às vezes, durante esse processo, ocorrem erros e pedaços únicos de DNA são codificados com as informações incorretas. Se compararmos duas cadeias de DNA e contarmos as diferenças entre elas, podemos ver quantos erros ocorreram. Isso é conhecido como "Distância de Hamming".

Lemos o DNA usando as letras C,A,G e T. Duas fitas podem se parecer com isso:

GA**G**C**C**TA**C**T**A**ACGG**GA**T
CA**T**C**G**TA**A**T**G**ACGG**CC**T
^ ^ ^ ^ ^ ^^

Eles têm 7 diferenças e, portanto, a distância de Hamming é 7.