

**PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO GRANDE DO SUL**  
Faculdade de Informática

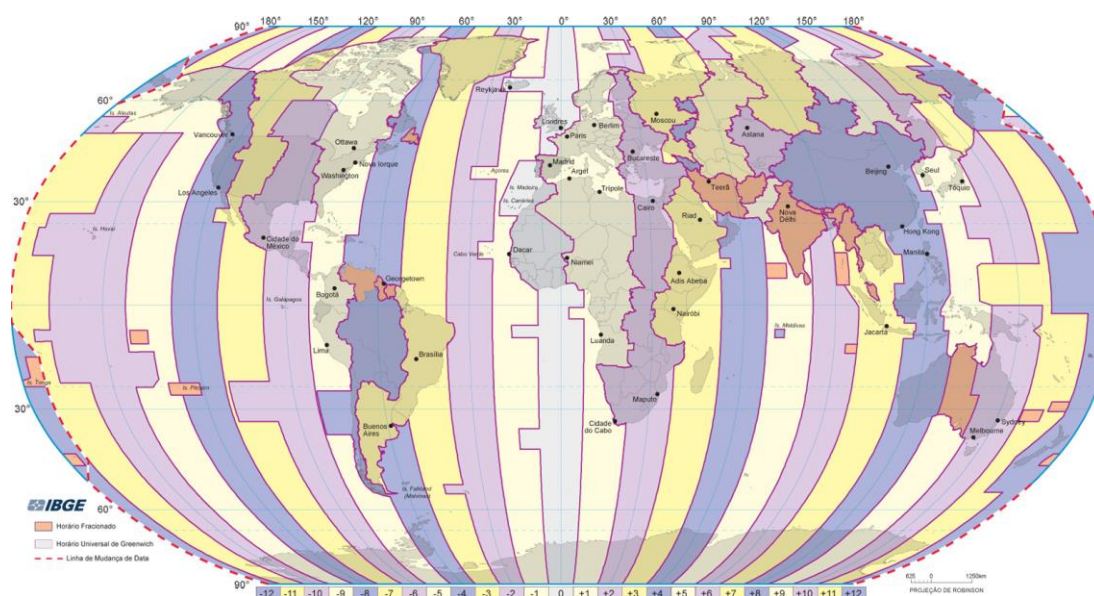
**Curso:** Engenharia de Software  
**Disciplina:** Fundamentos de Programação  
**Turma:** 127  
**Data:** 12/09/2016  
**Nome:** \_\_\_\_\_

G1: ☒ P1 ☐ P2 ☐ P3 ☐ PS  
☐ G2

**Nota:** \_\_\_\_\_ **Rubrica:** \_\_\_\_\_  
**Profa. Luana Müller**

Leia com **atenção** os enunciados: a interpretação das questões é parte da prova. Todos os programas devem ser codificados usando a Linguagem **Java** e respeitando as convenções estabelecidas em aula.

1. Ao fazer uma viagem para um país estrangeiro temos que lidar com questões relacionadas ao fuso horário. O mundo possui ao todo 24 fusos horários, estando 12 para cada lado do meridiano de Greenwich. Observe a figura a seguir:



Logo, por exemplo, quando aqui no Brasil são 10:00, são 16:00 em Moscou, onde a diferença de fuso é de +6 horas. Já no Hawaii, são 03:00, onde a diferença de fuso é de -7 em relação ao nosso.

- a. **(3.0 pts)** Você deve construir um programa que leia o horário de partida (horas e minutos), tempo de viagem (horas e minutos) e diferença do fuso horário; e apresente qual será o horário local no país de destino no momento da chegada.

OBS: Tenha cuidado com as horas acima de 24 e abaixo de 0. Elas devem ser tratadas.

Veja na tabela alguns exemplos de entradas e saídas esperadas:

Entradas					Saída
Horas origem	Minutos Origem	Horas Viagem	Minutos Viagem	Dif. Fuso	Hora chegada
10	20	8	10	2	20:30
01	10	1	15	-3	23:25
03	50	18	40	6	4:30

- b. **(1.0 pts)** Usando seu programa, apresente testes de mesa para os três exemplos de entradas descritos acima e verifique se a saída obtida é a mesma descrita na tabela.

2. O código abaixo implementa um problema matemático, descoberto por Collatz, em 1937. Ele descobriu que *qualquer número natural, quando aplicado a regra, no fim dará sempre 1*.

```
1 public class Collatz
2 {
3     public static void main(String [] args) {
4         int voltas = 0;
5         int n = 20;
6         while(n != 1) {
7             if( n % 2 == 0) {
8                 n = n/2;
9             } else {
10                 n = (3 * n) + 1;
11             }
12             voltas++;
13         }
14         System.out.println("Voltas: " + voltas);
15     }
16 }
```

- (1.0 pts)** Apresente o **teste de mesa** de forma a provar a regra para este valor  $n$  indicado no código.
  - (0.5 pts)** **Explique**, de forma objetiva, o que o algoritmo está fazendo.
  - (0.5 pts)** Se **movermos** a chave `}` da linha 11, para a linha 13, deixando assim duas chaves lado a lado `}}`, os resultados seriam afetados de alguma forma? Justifique sua resposta.
3. Construa um programa para apoio a contagem de votos da eleição para síndico de seu condomínio. Sabe-se que há 800 moradores e que todos irão votar. Sabe-se também que há 3 candidatos, representados pelos números 1, 2 e 3. Faça um programa de acordo com as especificações a seguir:
- (1.5 pts)** O programa deverá fazer a leitura da opção de voto de cada um dos moradores e contabilizar os votos de cada candidato. Votos em brancos são computados pelo número 4, e votos nulos, computados por qualquer valor diferente dos citados anteriormente.
  - (1.5 pts)** Seu programa deverá apresentar o total de votos nulos e de cada candidato e % destes votos sobre o total. De acordo com a legislação da cidade, votos em branco deverão ser divididos entre os candidatos de acordo com o % que cada um recebeu do total de votos.
  - (1.0 pts)** Ao final, verifique e imprima qual dos candidatos venceu a eleição.

*Boa prova!*