

## **RUS0001 - FUNDAMENTOS DE PROGRAMAÇÃO**

Primeira Avaliação Parcial – 05 de outubro de 2015

Professora Dr<sup>a</sup> Viviane Menezes

NOME DO ALUNO :
CURSO :
MATRICULA :

### **INSTRUÇÕES**

1. Preencha o cabeçalho acima. Não se esqueça da sua assinatura.
2. A prova deve ser feita sem consulta a computadores, apontamentos, cadernos, livros ou colegas.
3. As questões devem ser respondidas com canetas nas cores azul ou preta.

**DURAÇÃO DA PROVA: 120 minutos**

	Nota
Questão 1	
Questão 2	
Questão 3	
Questão 4	
Questão 5	
TOTAL	

**B O A   P R O V A !**

1. (valor 2.0 pontos) **FUP\$** é uma moeda do país **Gênios da Computação**. Considere que cada aluno de fundamentos de programação recebeu em sua conta bancária um valor em FUP\$, calculado de acordo com o seu número de matrícula. O cálculo é o seguinte:

- cada dígito da matrícula que é divisível por 3 vale 3 FUP\$
- cada dígito da matrícula que não é divisível por 3 vale 1 FUP\$.

O aluno *Alan Turing da Silva*, por exemplo, digitou sua matrícula **312346** e verificou que o programa imprimiu na tela o valor de 14 FUP\$. Faça um programa em C que **receba o número de matrícula**  $n$  de um aluno da UFC e imprima na tela:

- a) o valor em FUP\$ correspondente ao número de matrícula do aluno.
- b) o valor em R\$ a ser depositado na conta do aluno. Cada FUP\$ equivale a 2 R\$.

```
int main(){
    printf("Digite o seu número de matrícula: ");
    scanf("%d", &n);
    ...
}
```

2. (valor 2.0 pontos) Conheça essa sequência 1, 1, 2, 3, 5, 8, 13, 21, 34, ...? Cada número tem relação com seus dois últimos predecessores. Você deve escrever um programa em C que recebe um número  $n$  passado pelo usuário e informe na tela os  $n$  primeiros números dessa sequência.

3. (valor 2.0 pontos) Elabore um programa em C que receba um número  $n$  fornecido pelo usuário e calcule o seu **fatorial**. Se o número for negativo, informe que o valor é inválido. Sabemos que o fatorial de um número  $n$ , representando por  $n!$ , é dado por:

$$n \times (n - 1) \times (n - 2) \times \cdots \times 1 \quad \text{para } n > 0$$

e

$$n! = 1 \quad \text{para } n = 0.$$

4. (valor 2.0 pontos) **Números primos** são números naturais que têm apenas dois divisores: o 1 e ele mesmo. Faça um programa que informe **quantos primos** existem entre 0 e um número  $n$  (incluindo  $n$ ), passado pelo usuário. **Dica:** Use repetições encaixadas.

5. (valor 2.0 pontos) Dado um número inteiro  $n > 0$ , passado pelo usuário, faça um programa em C que informe na tela o valor da soma

$$s_n = \frac{1}{n} + \frac{2}{(n-1)} + \frac{3}{(n-2)} + \cdots + \frac{n}{1}$$