

## Oficina de Programação em C (ICP037)

Prof. Ronald Souza – IC/UFRJ

### Prova de Reposição – 18/07/2024

#### Questão 1) (1.0 ponto)

Que sequência o programa abaixo imprimirá? *Na sua resposta, basta informar o que será impresso. Não é preciso descrever o passo a passo.*

```
#include <stdio.h>
void sequencia(int x) {
    if (x % 10) {
        printf("%d", x);
        sequencia(x - 1);
        printf("%d", x);
    }
}
int main() {
    sequencia(2);
    return 0;
}
```

#### Questão 2)

- a) (1.0 ponto) Implemente a função **totalLetra**, que recebe uma **string** e também uma letra (**caractere**) qualquer e retorna um **inteiro** correspondente ao total de vezes que essa letra ocorre na string.
- b) (1.0 ponto) A função abaixo recebe um vetor de inteiros de tamanho N. Cada posição do vetor de entrada é **necessariamente** um único dígito entre 0 e 9. **Explique textualmente o que faz a função abaixo:**

```
void funcao(int vet[], int n) {
    int f[10] = { 0 }; //Zera todas as posições do vetor f.
    for (int i = 0; i < n; i++)
        f[vet[i]]++;
    for (int i = 0; i < 10; i++)
        printf("%d: %d\n", i, f[i]);
}
```

#### Questão 3 (1.0 ponto)

Deseja-se implementar uma função que retorna a **soma dos dígitos** de um número de entrada. Por exemplo, para o número 1021 a função retornará 4, pois  $1+0+2+1 = 4$ .

Foi iniciada a escrita da função, conforme o código abaixo:

```
int somaDigitos(int a) {
    // 0 que retornar?
}
```

→ No seu caderno de respostas (**e não aqui!**), indique claramente, **em letra de fôrma**, dentre as alternativas abaixo, a que contém o corpo da função que captura corretamente o comportamento acima descrito:

- A) return (a) ? a % 10 + somaDigitos(a / 10) : 0;
- B) return (a % 10) ? a % 10 + somaDigitos(a / 10) : 0;
- C) return (a == 0) ? 0 : somaDigitos(a / 10);
- D) return (a / 10) ? a % 10 + somaDigitos(a / 10) : 0;
- E) return (a) ? a / 10 + somaDigitos(a % 10) : a;

**Questão 4)** Trabalharemos com o Tipo Abstrato de Dado (TAD) **Cor**. Para isso, siga os seguintes passos:

a) **(1.0 ponto)** Crie o **tipo estruturado Cor**. Sua estrutura conterá 3 **inteiros**, **r**, **g** e **b**, referentes aos **canais** vermelho (*red*), verde (*green*) e azul (*blue*).

b) **(1.0 ponto)** No sistema RGB, o valor de cada um dos 3 canais deve **necessariamente** estar **entre 0 e 255** (inclusive), permitindo assim representarmos um total de  $256 * 256 * 256 = 16.777.216$  cores! Crie uma **função** chamada **nivel**, que realizará tal defesa. Especificamente, sua função **nivel** deve receber **um único inteiro a de entrada** e então retornar o **inteiro**:

(i) 0, se **a** < 0; (ii) 255, se **a** > 255; (iii) o próprio **a** para os demais casos (ou seja, quando  $0 \leq a \leq 255$ ).

c) **(1.5 ponto)** Agora, crie uma função que, dados 3 **inteiros**, **x**, **y** e **z** **quaisquer** de entrada, retorne uma **Cor** cujos canais **r**, **g** e **b** valem, respectivamente, **x**, **y** e **z**. **Defenda os valores dos canais**: o valor de todo canal deve estar entre 0 e 255 (inclusive), o que pode não ser o caso de **x**, **y** e **z**. *DICA: sua função do item anterior já faz isso!*

d) **(1.5 ponto)** Crie uma **função** que receba duas **cores** de entrada e retorne a cor correspondente à soma delas, onde cada canal é a soma dos respectivos canais das cores de entrada (i.e. **r** = soma dos vermelhos, **g** = soma dos verdes, **b** = soma dos azuis). *DICA: lembre-se que o valor máximo por canal é 255 e que você já possui uma função para tratar isso!* **Por exemplo:** Para 2 cores **c1** e **c2** de entrada, onde

**r1** = 0;      **g1** = 130; **b1** = 245 e

**r2** = 80;      **g2** = 100; **b2** = 55, a cor retornada terá os valores **r** = 80; **g** = 230; **b** = 255.

e) **(2.0 pontos)** Crie uma **função** que receba um vetor de **cores** e seu tamanho **N**, e então (i) ordena esse vetor em ordem **crescente** do canal **r** (vermelho) de cada cor pelo método **Bubble Sort** e (ii) imprime todas as cores no formato "(r, g, b)", **seguindo a ordenação crescente estabelecida no item (i)**. Sua função **não** deve retornar valores.

**Questão 5) (2.0 pontos)** O que será impresso ao final de cada iteração do programa abaixo? No seu caderno de respostas, faça uma tabela análoga à descrita a seguir, e a preencha com os valores esperados:

```
#include <stdio.h>
int main() {
    float a = 10, b = 2;
    int x = 0;
    for (int i = 1; i <= 3; i++) {
        b += i + i / 2;
        a = i + b / 2;
        x = b - a;
        if (b && x) {
            printf("a = %.1f; b = %.1f; x = %d\n", a, b, x);
            continue;
            b = 0;
        }
        a = 1 / 2;
        printf("a = %.1f; b = %.1f; x = %d\n", a, b, x);
    }
    return 0;
}
```

i	a	b	x
1	?	?	?
2	?	?	?
3	?	?	?