
MC102—Algoritmos e Programação de Computadores

Turmas U e V
Segundo Semestre de 2007
Prova 01
Prof. Luiz Fernando Bittencourt

18/09/2007

NOME : _____

RA : _____

ASSINATURA : _____

- Você deve fazer as questões 1, 2 e 5 e escolher UMA entre as questões 3 e 4. Então, **devem ser feitas exatamente 4 questões**, sendo as questões **1, 2 e 5 obrigatórias**. Note que a questão 3 vale mais que a questão 4, então fazendo a questão 3 a prova vale 11, fazendo a questão 4 a prova vale 10. Fazendo as duas, a prova vale 11 e a questão 4 não será corrigida.
- Só serão tiradas dúvidas relativas aos enunciados.
- Na correção será avaliado principalmente se o código e a lógica fazem **exatamente** o que foi pedido no enunciado. Também será avaliada a sintaxe da linguagem, principalmente uso das chaves nos blocos de comando, já que o esquecimento ou uso incorreto pode levar o programa a não fazer o que deveria. Note que não há implicações no esquecimento de um ou outro ponto e vírgula, mas poderá haver se houverem muitos comandos sem ponto e vírgula ou uso indevido/incorreto do ponto e vírgula. Outro ponto avaliado é a declaração de variáveis com tipos compatíveis com o uso. Por exemplo, se o enunciado diz que serão usados números reais mas os números são lidos como inteiros, será descontado. Ainda, se o enunciado diz que serão números inteiros e é usado float, também é passível de desconto. Obviamente há casos que declaração de tipo diferente do número lido é justificável, como por exemplo quando a variável recebe algum número não inteiro durante o programa. Nesse caso, deve haver justificativa do uso do tipo colocada como comentário no programa. Se houver questões que perguntam a saída de algum programa, a saída deve ser condizente com o tipo da variável.

1. (3.5) *Algoritmo do CPF*. Um número de CPF (Cadastro de Pessoa Física) válido é composto por 11 dígitos, sendo 9 dígitos independentes e 2 dígitos de verificação (DVs). Os dígitos de verificação servem para evitar erros ao copiar/digitar um CPF e para obtê-los são feitos cálculos utilizando os dígitos do CPF. Dessa forma, se houver um dígito errado no número do CPF, o cálculo dos dígitos verificadores resultará em números diferentes daqueles dos dígitos verificadores do CPF correto.

Calculando o primeiro dígito verificador do CPF:

O primeiro DV é calculado utilizando os 9 dígitos independentes, atribuindo-se um “peso” a cada dígito e utilizando um cálculo módulo 11. O peso do primeiro dígito é 10, do segundo é 9, do terceiro é 8, ..., do nono é 2. Um somatório da multiplicação dos pesos pelos dígitos é feito, e o resultado é dividido por 11. Se o resto dessa divisão é menor que 2, o primeiro dígito verificador será 0. Caso contrário, o resto dessa divisão é subtraído de 11, obtendo o primeiro dígito verificador. Exemplo: Seja o CPF 222.333.666-XY, onde XY são os dígitos verificadores que calcularemos. Temos:

digito	2	2	2	3	3	3	6	6	6
peso	10	9	8	7	6	5	4	3	2

Então, para o exemplo, o cálculo do primeiro dígito verificador seria:

- 1) $10*2+9*2+8*2+7*3+6*3+5*3+4*6+3*6+2*6=162$
- 2) O resto da divisão do resultado obtido (162) por 11 é 8.
- 3) Se esse resto fosse menor que 2, o primeiro dígito verificador seria 0.
- 4) Como não é o caso acima, subtrai-se o valor do resto de 11: $11-8=3$.
- 5) Primeiro dígito verificador é o número 3. O nosso CPF passa a ser 222.333.666-3Y

Calculando o segundo dígito verificador do CPF:

O cálculo do segundo dígito é semelhante ao do primeiro. A diferença é que os pesos de cada dígito são acrescidos de 1 e o primeiro DV é incluído no cálculo. Então temos:

digito	2	2	2	3	3	3	6	6	6	-	3
peso	11	10	9	8	7	6	5	4	3	-	2

E o cálculo fica:

- 1) $11*2+10*2+9*2+8*3+7*3+6*3+5*6+4*6+3*6+2*3=201$
- 2) O resto da divisão do resultado obtido (201) por 11 é 3.
- 3) Se esse resto fosse menor que 2, o segundo dígito verificador seria 0.
- 4) Como não é o caso acima, subtrai-se o valor do resto de 11: $11-3=8$.
- 5) Segundo dígito verificador é o número 8. O nosso CPF passa a ser 222.333.666-38

Transforme o algoritmo acima em um código em linguagem C que recebe como entrada 9 números inteiros de 1 dígito cada e calcula os dois dígitos verificadores do CPF. **ATENÇÃO: USO OBRIGATÓRIO DE ESTRUTURA DE REPETIÇÃO (FOR ou WHILE**

ou DO-WHILE) PARA O CÁLCULO DOS DVs E VETOR PARA LER O CPF. NÃO DEVE SER UTILIZADA UMA VARIÁVEL PARA CADA DÍGITO DO CPF. Assuma que o usuário sempre digita corretamente 9 números de 1 dígito cada, não sendo necessárias verificações da entrada. Exemplo de entrada e saída:

ENTRADA:

2 2 2 3 3 3 6 6 6

SAÍDA:

3 8

2. (2.0) Quais as saídas do código abaixo para as entradas listadas?

```
#include<stdio.h>

struct s {
    int a,b;
    float f;
};

int main(void)
{
    struct s m,p;

    scanf("%d", &m.a);
    m.b=m.a*10;

    p=m;
    p.f=p.a+p.b;

    while(p.f > 2.0) {
        if((p.f > 100.0) && (m.a == p.a))
            printf("Saída 1\n");
        if(p.f <= 5.0)
            printf("Saída 2\n");
        p.f=p.f/2;
    }
}
```

Entrada	Saída(s)	Entrada	Saída(s)
5		2	
-3		20	
10		0	

3. (2.5) Um número é primo se ele é divisível apenas por 1 e por ele mesmo. Para um número inteiro $K > 2$ ser primo é necessário e suficiente que K não seja divisível por nenhum inteiro maior que 1 e menor ou igual ao piso da raiz quadrada de K . Sabemos, ainda, que nenhum número par maior que 2 é primo. Sabendo que se K não é primo então $\exists y \mid y \text{ MOD } K = 0, 2 \leq y \leq \lfloor \sqrt{K} \rfloor$, escreva um programa em linguagem C que recebe um número inteiro como entrada e imprime na tela uma mensagem dizendo se o número é primo ou não. Considere que os números 1 e 2 são primos. Para calcular a raiz quadrada de um número X , use $\text{sqrt}(X)$.

4. (1.5) Escreva um programa em linguagem C que recebe como entrada 4 números inteiros e verifica se esses números formam uma P.A. (progressão aritmética). Se sim, o programa deve imprimir a razão da P.A. Se não, deve imprimir uma mensagem na tela.

5. (2.0) Escreva um programa em C que recebe como entrada uma string de tamanho máximo 100 e conta quantos caracteres dessa string são vogais e quantos são consoantes. Caracteres que não são letras não devem ser contados. Dados da tabela ASCII: 'A'=65, 'Z'=90, 'a'=97, 'z'=122.