

Algoritmos e Programação

Prova 2 – 30/06/2008

Prof. Marcus Vinícius Midena Ramos

Engenharia de Computação – UNIVASF

Questão 1 (2 pontos):

Codifique um programa que receba como entrada um número inteiro n e mostre na saída todos os divisores inteiros de n . Por exemplo, se a entrada for “20”, a saída deverá ser “1 2 4 5 10 20”.

```
#include <stdio.h>
#include <stdlib.h>
int main () {
    int n,i;
    scanf ("%d",&n);
    for (i=1;i<=n;i++)
        if (n%i==0) printf ("Divisível por %d\n",i);
    system ("pause");
}
```

Questão 2 (3 pontos):

Codifique as funções conforme especificação abaixo:

- a. (1 ponto) Função *strlen2 ()*:

Deve receber como parâmetros duas cadeias de caracteres: $s1$ e $s2$;
Deve retornar o maior comprimento entre as cadeias $s1$ e $s2$.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int strlen2 (char s1 [], char s2[]) {
    if (strlen (s1)>strlen (s2)) return strlen(s1);
    else return strlen (s2);
}
int main () {
    char x [80],y [80];
    gets (x);
    gets (y);
    printf ("O maior comprimento é %d\n", strlen2 (x,y));
    system ("pause");
}
```

- b. (1 ponto) Função *strstr2 ()*:

Deve receber como parâmetros três cadeias de caracteres: $s1$, $s2$ e $s3$;
Deve retornar 1 se $s1$ contém as subcadeias $s2$ e $s3$ ou 0 caso contrário.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int strstr2 (char s1 [], char s2[], char s3 []) {
    if (strstr (s1,s2)&&strstr (s1,s3)) return 1;
    else return 0;
}
```

```

int main () {
    char x[80],y[80],z[80];
    gets (x);
    gets (y);
    gets (z);
    printf ("Resultado de strstr2: %d\n", strstr2 (x,y,z));
    system ("pause");
}

```

c. (1 ponto) Função *strcnp2 ()*:

Deve receber como parâmetros três cadeias de caracteres: *s1*, *s2* e *s3*;

Deve retornar 1 se a concatenação da cadeia *s1* com a cadeia *s2* coincidir com a cadeia *s3*, ou ainda se a cadeia *s1* coincidir com a concatenação da cadeia *s2* com a cadeia *s3*.

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int strcnp2 (char s1 [], char s2[], char s3 []) {
    char s12[160], s23[160];
    strcpy (s12,s1);
    strcat (s12,s2);
    strcpy (s23,s2);
    strcat (s23,s3);
    if ((strcmp (s1,s23)==0)|| (strcmp (s12,s3)==0)) return 1;
    else return 0;
}
int main () {
    char x[80],y[80],z[80];
    gets (x);
    gets (y);
    gets (z);
    printf ("Resultado de strcnp2: %d\n", strcnp2 (x,y,z));
    system ("pause");
}

```

Questão 3 (2 pontos):

(0,5 ponto) Codifique uma função denominada *primo* com as seguintes características:

1. A função deve aceitar como entrada um valor inteiro *n*;
2. A função deve retornar o valor 0 ou 1, da seguinte forma:
 - a. Se *n* é primo, então retorna 1;
 - b. Se *n* não é primo, então retorna 0.

```

int primo (int n) {
    int i,s;
    s=0;
    for (i=2;i<=n-1;i++)
        if (n%i==0) s++;
    if (s) return 0;
    else return 1;
}

```

(0,5 ponto) Codifique uma função denominada *ímpar* com as seguintes características:

1. A função deve aceitar como entrada um valor inteiro *n*;
2. A função deve retornar o valor 0 ou 1, da seguinte forma:
 - a. Se *n* é ímpar, então retorna 1;
 - b. Se *n* é par, então retorna 0.

```

int impar (int n) {
    if (n%2==0) return 0;
    else return 1;
}

```

(1 ponto) Codifique um programa que aceita como entrada um número inteiro x e determina quantos números primos ímpares existem entre x (inclusive) e x^2 (inclusive). Devem ser usadas as duas funções acima.

```

#include <stdio.h>
#include <stdlib.h>
int main () {
    int n,i;
    scanf ("%d",&n);
    for (i=n;i<=n*n;i++)
        if (primo (i) && impar (i)) printf ("Primo e ímpar: %d\n", i);
    system ("pause");
}

```

Questão 4 (3 pontos):

Codifique uma função que aceita como entrada uma matriz de inteiros 10x10 (10 linhas e 10 colunas) e gere como saída uma outra matriz de inteiros 10x10 em que as colunas da segunda matriz correspondam às linhas da primeira matriz.

```

#include <stdio.h>
#include <stdlib.h>
#define SIZE 3
void ler (int m [SIZE][SIZE]) {
    int i,j;
    for (i=0; i<SIZE;i++)
        for (j=0;j<SIZE;j++)
            scanf ("%d", &m[i][j]);
}
void trocar (int m [SIZE][SIZE], int n [SIZE][SIZE]) {
    int i,j;
    for (i=0; i<SIZE;i++)
        for (j=0;j<SIZE;j++)
            n[j][i]=m[i][j];
}
void escrever (int m [SIZE][SIZE]) {
    int i,j;
    printf ("\n");
    for (i=0; i<SIZE;i++) {
        for (j=0;j<SIZE;j++)
            printf ("%d ", m[i][j]);
        printf ("\n");
    }
}
int main () {
    int a [SIZE][SIZE];
    int b [SIZE][SIZE];
    ler (a);
    escrever (a);
    trocar (a,b);
    escrever (b);
    system ("pause");
}

```