

# INTRODUÇÃO À PROGRAMAÇÃO

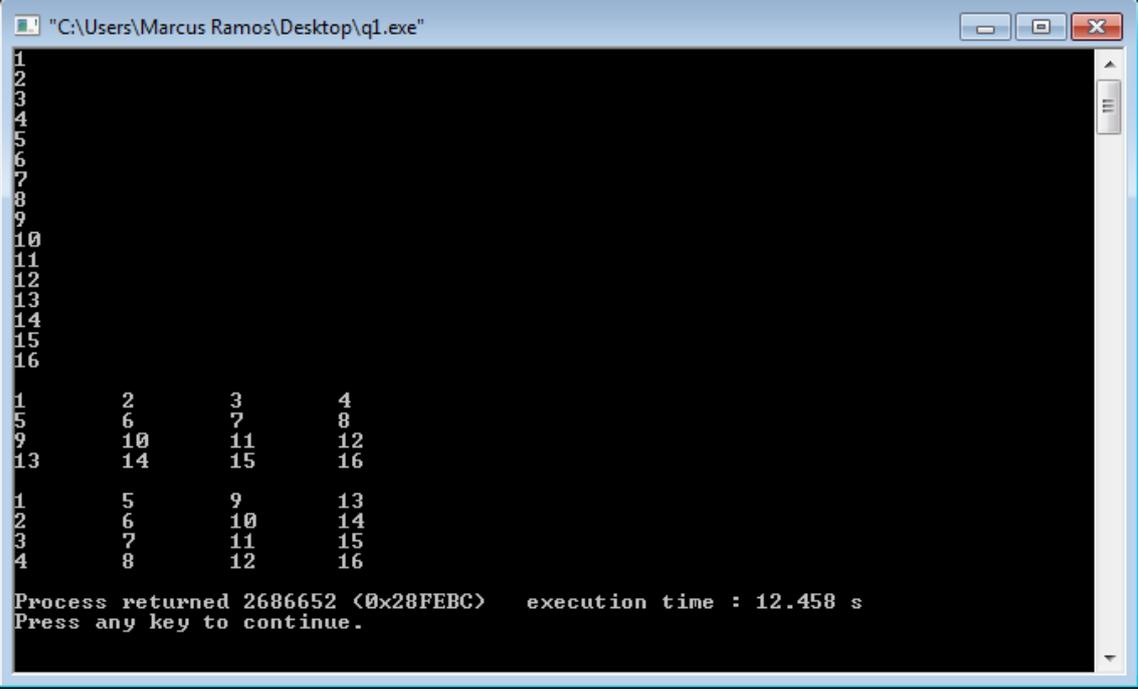
Prova 2 – Prof. Marcus Ramos – 25 de abril de 2013

Questão 1 (2,5 pontos):

Faça um programa que copie as linhas de uma matriz quadrada de números inteiros “m” para as colunas de uma outra matriz “n”, com as mesmas dimensões.

- Utilizar #define para especificar o tamanho da matriz;
- O programa deve inicialmente ler dados do teclado e preencher a matriz “m”;
- A primeira linha de “m” deve ser copiada para a primeira coluna de “n”, a segunda linha para a segunda coluna, e assim sucessivamente;
- Ao final, o programa deve imprimir na tela o conteúdo de “m” e de “n”, na forma de linhas e colunas.

Exemplo para uma matriz 4x4:



```
"C:\Users\Marcus Ramos\Desktop\q1.exe"
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
1      2      3      4
5      6      7      8
9      10     11     12
13     14     15     16
1      5      9      13
2      6      10     14
3      7      11     15
4      8      12     16
Process returned 2686652 (0x28FEBC)   execution time : 12.458 s
Press any key to continue.
```

```
#include <stdio.h>
#define Q 4
int main () {
int m[Q][Q], n[Q][Q];
int i,j;
for (i=0;i<Q;i++)
    for (j=0;j<Q;j++)
        scanf ("%d",&m[i][j]);
printf ("\n");
for (i=0;i<Q;i++) {
    for (j=0;j<Q;j++)
        printf ("%d\t",m[i][j]);
    printf ("\n");
}
```

```

printf ("\n");
for (i=0;i<Q;i++)
    for (j=0;j<Q;j++)
        n[j][i]=m[i][j];
for (i=0;i<Q;i++) {
    for (j=0;j<Q;j++)
        printf ("%d\t",n[i][j]);
    printf ("\n");
}
}

```

Questão 2 (2,5 pontos):

Suponha que “a”, “b” e “c” sejam os coeficientes de uma equação quadrática. Complete o programa abaixo para que ele calcule o discriminante da mesma ( $b^2-4*a*c$ ), acessando os valores de “a”, “b” e “c” exclusivamente através das variáveis “a2”, “b2” e “c2”.

Exemplo de entradas e saída:

```

"C:\Users\Marcus Ramos\Desktop\q2.exe"
3
2
1
0 discriminante para os coeficientes 3, 2 e 1 eh -8.
Process returned 54 (0x36) execution time : 2.277 s
Press any key to continue.

```

```

#include <stdio.h>
int main () {
float a,b,c,discrim;
float *a1,*b1,*c1;
float **a2,**b2,**c2;
scanf ("%f",&a);
scanf ("%f",&b);
scanf ("%f",&c);
...
}

```

```

#include <stdio.h>
int main () {
float a,b,c,discrim;
float *a1,*b1,*c1;
float **a2,**b2,**c2;
scanf ("%f",&a);
scanf ("%f",&b);
scanf ("%f",&c);
a1=&a;
b1=&b;
c1=&c;
a2=&a1;
b2=&b1;
c2=&c1;
}

```

```

discrim=(**b2)*(**b2)-4*(**a2)*(**c2);
printf ("\nO discriminante para os coeficientes %1.f, %1.f e
%1.f eh %1.f.\n",a, b, c, discrim);
}

```

Questão 3 (2,5 pontos):

Construa um programa com três funções, respectivamente denominadas “raizes”, “raiz1” e “raiz2”, que serão usadas para calcular as raízes de uma equação do segundo grau com coeficientes reais.

A função “raizes” deve aceitar como parâmetros três valores “a”, “b” e “c” do tipo float, correspondentes aos coeficientes de uma equação quadrática, e retornar como resultado um valor inteiro indicando se a mesma possui duas soluções reais distintas, apenas uma solução real, ou se não possui soluções reais. Uma equação do segundo grau possui duas soluções reais distintas quando o discriminante é maior que zero, uma única solução real quando o discriminante é zero e não possui soluções reais quando o discriminante é negativo). A função deve retornar o valor 2 se a equação possuir duas raízes reais distintas, 1 se possui apenas uma raiz real ou 0 se não possuir raízes reais.

As funções “raiz1” e “raiz2” devem aceitar como parâmetros três valores “a”, “b” e “c” do tipo float, correspondentes aos coeficientes de uma equação quadrática, e retornar como resultado a raiz correspondente, supondo que a equação possua pelo menos uma raiz no caso da primeira função, e duas raízes no caso da segunda.

Exemplifique o uso das três funções num programa que faz a leitura de valores digitados pelo usuários e informa sobre a quantidade de raízes existentes, indicando quais elas são em cada caso.

As raízes de uma equação de segundo grau são calculadas através da fórmula:  
 $(-b (+/-) \text{raizquadrada}(b^2 - 4*a*c)) / (2*a)$

Exemplos de entradas e respectivas saídas:

```

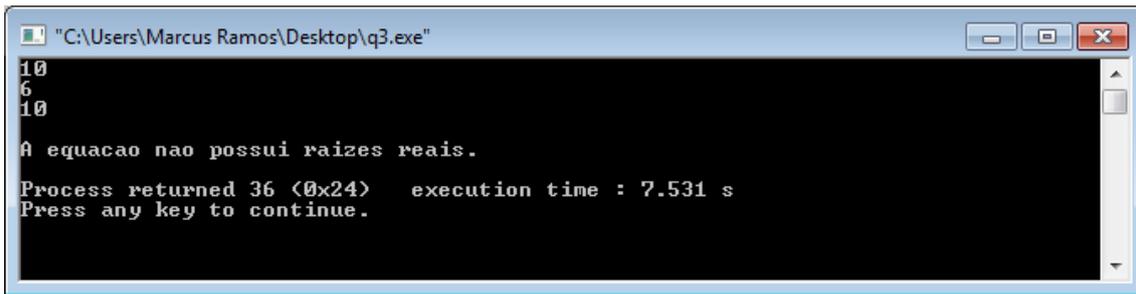
"C:\Users\Marcus Ramos\Desktop\q3.exe"
1
-2
-3
As raizes sao 3.000000 e -1.000000.
Process returned 37 (0x25)   execution time : 7.224 s
Press any key to continue.

```

```

"C:\Users\Marcus Ramos\Desktop\q3.exe"
1
8
16
A raiz eh -4.000000.
Process returned 22 (0x16)   execution time : 7.741 s
Press any key to continue.

```



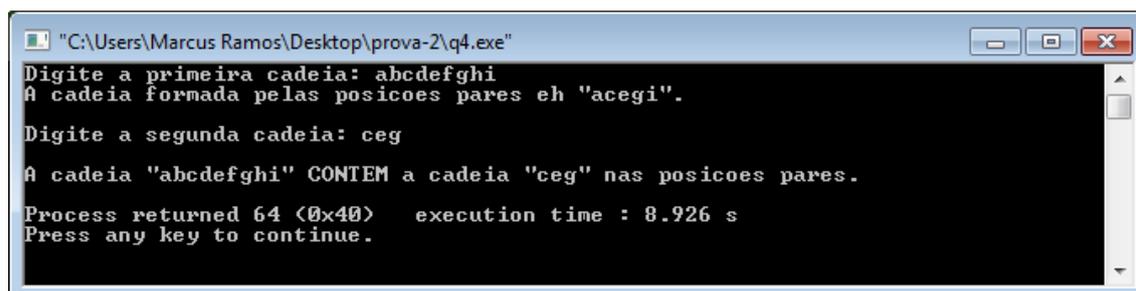
```
"C:\Users\Marcus Ramos\Desktop\q3.exe"
10
6
10
A equacao nao possui raizes reais.
Process returned 36 (0x24)   execution time : 7.531 s
Press any key to continue.
```

```
#include <stdio.h>
#include <math.h>
int raizes (float a, float b, float c) {
float delta=b*b-4*a*c;
if (delta==0) return 1;
    else if (delta>0) return 2;
    else return 0;
}
float raiz1 (float a, float b, float c) {
return (-b+sqrt(b*b-4*a*c))/(2*a);
}
float raiz2 (float a, float b, float c) {
return (-b-sqrt(b*b-4*a*c))/(2*a);
}
int main () {
float a,b,c;
scanf ("%f",&a);
scanf ("%f",&b);
scanf ("%f",&c);
if (raizes (a,b,c)==2) printf ("\nAs raizes sao %f e
%f.\n",raiz1(a,b,c),raiz2(a,b,c));
    else if (raizes (a,b,c)==1) printf ("\nA raiz eh
%f.\n",raiz1(a,b,c));
    else printf ("\nA equacao nao possui raizes reais.\n");
}
```

Questão 4 (2,5 pontos):

Faça um programa que aceita como entrada um string "s" com até 100 caracteres, um segundo string "t" com até 50 caracteres, e verifica se o string "t" está contido no string que é formado apenas pelos caracteres localizados nas posições pares do string "s".

Exemplos de entradas e respectivas saídas:



```
"C:\Users\Marcus Ramos\Desktop\prova-2\q4.exe"
Digite a primeira cadeia: abcde fghi
A cadeia formada pelas posicoes pares eh "acegi".
Digite a segunda cadeia: ceg
A cadeia "abcde fghi" CONTEM a cadeia "ceg" nas posicoes pares.
Process returned 64 (0x40)   execution time : 8.926 s
Press any key to continue.
```

```
"C:\Users\Marcus Ramos\Desktop\prova-2\q4.exe"
Digite a primeira cadeia: abcdefgh
A cadeia formada pelas posicoes pares eh "aceg".

Digite a segunda cadeia: ae

A cadeia "abcdefgh" NAO CONTEM a cadeia "ae" nas posicoes pares.

Process returned 66 (0x42)   execution time : 15.374 s
Press any key to continue.
```

```
#include <stdio.h>
#include <string.h>
int main (){
char s[100];
char spar[50];
char t[50];
int i,j;
printf ("Digite a primeira cadeia: ");
gets (s);
j=0;
for (i=0;i<strlen(s);i=i+2) {
    spar[j]=s[i];
    j++;
}
if (strlen(s)%2==0) {
    spar[j]=s[i+1];
    spar[j+1]='\0';
}
else spar[j]='\0';
printf ("A cadeia formada pelas posicoes pares eh \"%s\".\n\n",
spar);
printf ("Digite a segunda cadeia: ");
gets (t);
if (strstr (spar,t)) printf ("\nA cadeia \"%s\" CONTEM a cadeia
\"%s\" nas posicoes pares.\n",spar,t);
else printf ("\nA cadeia \"%s\" NAO CONTEM a cadeia \"%s\" nas
posicoes pares.\n",spar,t);
}
```