

Nome:			RA:				
Assinatura:	1	2	3	4			
MC102 – GABARITO Prova 1 2009b Prof. Rogério Drummond	2	3	3	3			

Use lapis de preferência. Desenhe um retângulo envolta de cada resposta, identificando claramente a questão respondida. Só serão consideradas as respostas nos locais indicados. Nenhum outro papel além desta prova é permitido durante a aplicação deste teste. Calculadoras, telefones, radios, etc também não são permitidos.

Funções auxiliares:

```
float sqrtf(float x);
long double sqrtl(long double x);
float powf(float x, float y);
long long proximoPrimo(long long n); // retorna o menor primo maior que n
```

1. Implemente a função `float calculap(float a, float b, float c, float x)` que retorna o valor do do polinômio ax^2+bx+c no ponto x .

```
float calculap(float a, float b, float c, float x) {
    return a*x*x + b*x + c;
}
```

2. Implemente a função `void raizp(float a, float b, float c)` que imprime as raízes do polinômio ax^2+bx+c . No caso de raiz dupla, imprima 2 vezes o mesmo número. Você pode supor que o delta de $p(x)$ é sempre maior que zero.

```
void raizp(float a, float b, float c) {
    float delta = b*b - 4*a*c;
    printf("%f %f", (-b + sqrtf(delta))/(2*a), (-b - sqrtf(delta))/(2*a));
}
```

3. Implemente um programa que apresenta o menu abaixo e execute os comandos associados. Não é necessário ler os valores de a , b , c e x .

0-Sair

1-Calcula $p(x)$

2-Imprime as raízes de $p(x)$

Seu programa deve continuar a apresentar o menu até que a opção 0 seja escolhida.

```
#include <stdio.h>
#include <math.h>
int main(void) {
    int opcao;
    do {
        // imprime menu e le opcao
        do {
            printf("0-Sair/n1-Calcula p(x)\n2-Imprime as riazes de p(x)\n");
            printf("Escolha uma das opcoes acima: ");
            scanf("%d", &opcao);

            // valida opcao
            if (opcao < 0 || opcao > 2)
                printf("\n*** As opcoes validas sao 0, 1 e 2. Tente novamente. ***\n");
        } while (opcao < 0 || opcao > 2);

        // executa opcao escolhida
        if (opcao == 1)
            printf("\n0 valor de p(%f) eh: %f\n", x, calculap(a, b, c, x));
        else if (opcao == 2)
            raizp(a, b, c);
    } while (opcao != 0);
    return 0;
}
```

4. Dois números são coprimos se o único divisor comum entre eles é 1. Implemente a função `int coprimo(long long a, long long b)` que retorna 1 se a e b são coprimos ou 0 caso contrário. Você pode usar a função `long long proximoPrimo(long long n)` e `long double sqrtl(long double x)`.

```
int coprimo(long long a, long long b) {
    long long p = 2, menor;
    if (a==b)
        return 0;
    if (a < b)
        menor = a;
    else
        menor = b;

    if (a%b == 0 || b%a == 0)
        return 0;
    for (; p <= sqrtl(menor); p = proximoPrimo(p))
        if (a%p == 0 && b%p == 0)
            return 0;
    return 1;
}
```