

CÓDIGO ASSEMBLY

section .data

```
a dd 2.0    ; coeficiente 'a'
b dd -5.0   ; coeficiente 'b'
c dd 2.0    ; coeficiente 'c'
```

section .bss

```
delta resd 1 ; variável para armazenar o delta
x1 resd 1    ; raiz 1
x2 resd 1    ; raiz 2
```

section .text

```
global _start
```

_start:

```
; Calcula delta:  $\Delta = b^2 - 4ac$ 
fld dword [b]    ; carrega 'b' para o topo da pilha FPU
fmul dword [b]   ; multiplica 'b' por 'b'
fld dword [a]    ; carrega 'a' para o topo da pilha FPU
fld dword [c]    ; carrega 'c' para o topo da pilha FPU
fmul dword [a]   ; multiplica 'a' por 'c'
fmul dword dword 4.0 ; multiplica por 4
fsubp           ; subtrai o produto de 'a' e 'c' multiplicado por 4 do produto de 'b' e 'b'
fstp dword [delta] ; armazena o resultado em 'delta'
```

; Calcula as raízes:

```
cmp dword [delta], 0 ; verifica se delta é zero
jle raizes_complexas ; se delta for menor ou igual a zero, raízes complexas
```

; raízes reais

```
fld dword [b]    ; carrega 'b' para o topo da pilha FPU
fchs             ; muda o sinal de 'b' (b negativo)
fadd dword [delta] ; soma delta a 'b'
fsqrt            ; calcula a raiz quadrada
```

```
fld dword [a] ; carrega 'a' para o topo da pilha FPU
fadd dword [a] ; soma 'a' a raiz
fdiv ; divide a soma por 2*a
fstp dword [x1] ; armazena a primeira raiz
```

```
fld dword [b] ; carrega 'b' para o topo da pilha FPU
fchs ; muda o sinal de 'b' (b negativo)
fsub dword [delta] ; subtrai delta de 'b'
fsqrt ; calcula a raiz quadrada
fld dword [a] ; carrega 'a' para o topo da pilha FPU
fadd dword [a] ; soma 'a' a raiz
fdiv ; divide a soma por 2*a
fstp dword [x2] ; armazena a segunda raiz
```

; Saída

```
mov eax, 1 ; syscall para 'exit'
xor ebx, ebx ; código de saída 0
int 0x80 ; interrupção do sistema
```

raizes_complexas:

```
; manipulação de raízes complexas
; (não implementado neste exemplo)
jmp fim
```

fim:

```
mov eax, 1 ; syscall para 'exit'
xor ebx, ebx ; código de saída 0
int 0x80 ; interrupção do sistema
```

CÓDIGO ORIGINAL C#

```
using System;

class Program
{
    static void Main(string[] args)
    {
        double a = 2.0;
        double b = -5.0;
        double c = 2.0;

        // Calcula o delta:  $\Delta = b^2 - 4ac$ 
        double delta = b * b - 4 * a * c;

        if (delta >= 0)
        {
            // raízes reais
            double x1 = (-b + Math.Sqrt(delta)) / (2 * a);
            double x2 = (-b - Math.Sqrt(delta)) / (2 * a);

            // Imprime as raízes
            Console.WriteLine("Raiz 1: " + x1);
            Console.WriteLine("Raiz 2: " + x2);
        }
        else
        {
            Console.WriteLine("As raízes são complexas.");
        }
    }
}
```