Tradução C / Assembly – Exemplo 2

```
float fun(float, int);

void main(void)
{
   float  res;

   res = fun( 12.5E-2, 2 );
   print_float( res );  // syscall 2
}
```

```
float fun(float a, int m)
{
   float val;
   if( a >= -5.6 )
      val = (float)m * (a - 32.0);
   else
      val = 0.0;
   return val;
}
```

Tradução C / Assembly – Exemplo 2

S

```c
void main(void)
{
    float  res;

    res = fun( 12.5E-2, 2 );
    print_float( res );      // syscall 2
}
```

```c
float fun(float a, int k)
```

```
        .data
k1:     .float 12.5E-2          # 12.5 x 10^-2
k2:     .float -5.6
k3:     .float 32.0
k4:     .float 0.0
        .text
        .globl main             # res: $f12
main:   ...                     # salvaguarda $ra
        l.s     $f12, k1        # $f12 = 12.5E-2
        li      $a0, 2          # $a0 = 2
        jal     fun             #
        mov.s   $f12, $f0       # res = fun(12.5E-2, 2)
        li      $v0, 2          #
        syscall                 # print_float(res)
        ...                     # repõe $ra
        jr      $ra             #
```

Tradução C / Assembly – Exemplo 2

```
float fun(float a, int m)
{
    float val;
    if( a >= -5.6)
        val = (float)m * (a - 32.0);
    else
        val = 0.0;
    return val;
}
```

```
        .data
k1:     .float 12.5E-2
k2:     .float -5.6
k3:     .float 32.0
k4:     .float 0.0
```

```
# val: $f2   / a: $f12   /   m: $a0

fun:    l.s      $f0, k2          # $f0 = -5.6
        c.lt.s   $f12, $f0        # if( a >= -5.6 )
        bc1t     else             # {
        l.s      $f2, k3          #     val = 32.0
        sub.s    $f2, $f12, $f2   #     val = a - 32.0
        mtc1     $a0, $f0         #     $f0 = m
        cvt.s.w  $f0, $f0         #     $f0 = (float)m
        mul.s    $f2, $f0, $f2    #     val = (float)m * val
        j        endif            # } else
else:   l.s      $f2, k4          #     val = 0.0
endif:  mov.s    $f0, $f2         # return val;
        jr       $ra              #
```