```c
#include <stdio.h>

float sum( float, float );

int main()
{
    float first = 1.0;
    float last = 10.0;
    float total;

    total = sum( first, last );

    printf( "Total: %6.1f\n", total );

    return 0;
}
```

```assembly
.text
.balign 4
one:   .float  1.0
zero:  .float  0.0

.balign 4

sum:
    push    {lr}
    fcpys   s2, s0          @ Copy 1st arg (first value in seq)
    fcpys   s3, s1          @ Copy 2nd arg (last value in seq)
    ldr     r0, =zero       @ Get address of 0.0
    flds    s0, [r0]        @ Load 0.0 (sum of all values)
    ldr     r1, =one        @ Get address of 1.0
    flds    s1, [r1]        @ Load 1.0 (initial value in seq)

loop:
    fcmps   s2, s3          @ Compare current to limit
    fmstat                           @ Copy NZCV from FPSCR to CPSR
    bgt     endloop
    fadds   s0, s0, s2      @ Add current value to sum
    fadds   s2, s2, s1      @ Increment current value
    b       loop

endloop:
    pop     {lr}
    bx      lr
```
<3 lemon:˜/Examples > gcc example37.driver.c example37.support.s

<4 lemon:˜/Examples > a.out

Total: 55.0

<5 lemon:˜/Examples > gdb a.out

(gdb) break sum
Breakpoint 1 at 0x10484

(gdb) break endloop
Breakpoint 2 at 0x104b4

(gdb) run
Breakpoint 1, 0x00010484 in sum ()

(gdb) info float
fpscr 0x0
s0 1 (raw 0x3f800000)
s1 10 (raw 0x41200000)
s2 0 (raw 0x00000000)
s3 0 (raw 0x00000000)
.
.
s30 0 (raw 0x00000000)
s31 0 (raw 0x00000000)

(gdb) continue
Breakpoint 2, 0x000104b4 in endloop ()

(gdb) info float
fpscr 0x20000000
s0 55 (raw 0x425c0000)
s1 1 (raw 0x3f800000)
s2 11 (raw 0x41300000)
s3 10 (raw 0x41200000)
.
.
s30 0 (raw 0x00000000)
s31 0 (raw 0x00000000)

(gdb) continue
Continuing.

Total: 55.0

(gdb) q