

Sumar N enteros con signo de 32 bits en una plataforma de 32bits

Código

```
.section .data
    .macro linea
        #.int -1,-1,-1,-1
        #.int 1,-2,1,-2
        #.int 1,2,-3,-4
        #.int -1,-1,-1,-1
        #.int 0x7fffffff,0x7fffffff,0x7fffffff,0x7fffffff
        #.int 0x80000000,0x80000000,0x80000000,0x80000000
        #.int 0x04000000,0x04000000,0x04000000,0x04000000
        #.int 0xfc000000,0xfc000000,0xfc000000,0xfc000000
        #.int 0xf8000000,0xf8000000,0xf8000000,0xf8000000
        #.int 0xf0000000,0xe0000000,0xe0000000,0xd0000000
    .endm

lista:    .irpc i,12345678
            linea
        .endr

longlista: .int (.-lista)/4
resultado: .quad 0x012345678ABCDEF
formato:   .ascii "suma = %lld = %llx hex\n\0" #formato para 64 bits

.section .text
#_start:   .global _start
main: .global main
    mov $lista, %ebx
    mov longlista, %ecx
    call suma
    mov %eax, resultado
    mov %edx, resultado+4
    push resultado+4
    push resultado
    push resultado+4
    push resultado
    push $formato
    call printf
    add $20, %esp
    mov $1, %eax
    mov $0, %ebx
    int $0x80

suma:
    mov $0, %edi
    mov $0, %ebp
    mov $0, %esi

bucle:
    mov (%ebx,%esi,4), %eax
    cld
    add %eax, %edi
    adc %edx, %ebp
    inc %esi
    cmp %esi,%ecx
    jne bucle
    mov %edi, %eax
    mov %ebp, %edx
    ret
```

Tabla de resultados

Valores	Resultado
-1...	0
1, -2...	-16
1,2,-3,-4...	-32
0x7fffffff...	FFFFFFE0
0x80000000...	FFFFFFF000000000
0x04000000...	800000000
0x08000000...	FFFFFFE0
0xfc000000	FFFFFFFF80000000
0xf8000000	FFFFFFFF00000000
0xf0000000, 0xe0...,0xe0...,0xd0...,0xf0...	FFFFFFFC00000000