Sesión de suma64signed.s

1-.¿Cual es el maximo entero positivo que puede representarse (escribirlo en hexadecimal)? Si se sumaran los N=32 elementos de la lista inicializados a ese valor ¿que resultado se obtendría (en hexadecimal)? ¿Que valor aproximado tienen el elemento y la suma (indicarlo en multiplos de potencias binarias, ki, Mi, Gi)? Comprobarlo usando ddd

El máximo entero positivo que se puede representar es el 0x7FFFFFFF. La suma obtenida de los 32 elementos con el valor anterior es la siguiente:

```
juanka1995@juanka-laptop ~/practicas/ec/5.2 $ ./suma_con_signo
suma = fffffffe0
```

La suma 0xFFFFFFFe0 ≈ 2^{36} y el elemento 0x7FFFFFFF ≈ 2^{31}.

2-.Misma pregunta respecto a negativos: menor valor negativo en hexadecimal, suma, valores decimales aprox., usar ddd

El numero más pequeño que puede representarse en hexadecimal es el 0x80000000. La suma obtenida de los 32 elementos inicializados a ese número es de:

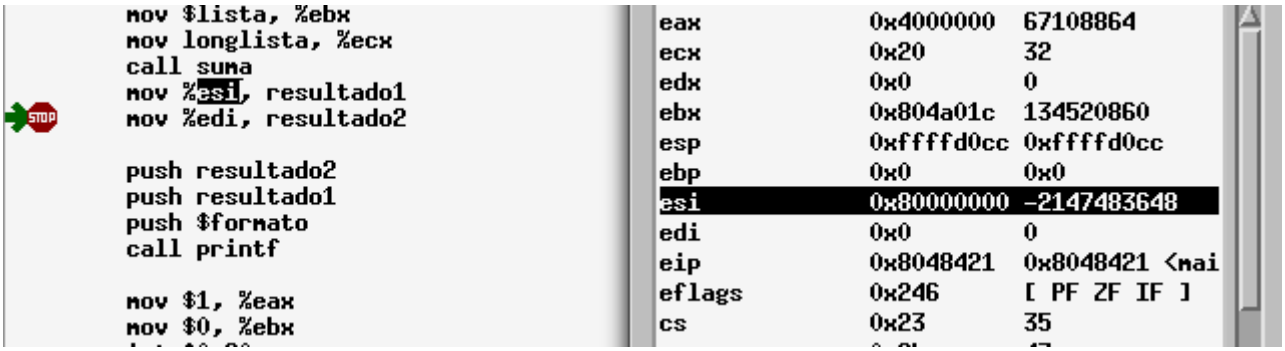
```
juanka1995@juanka-laptop ~/practicas/ec/5.2 $ ./suma_con_signo
suma = ffffffff000000000
```

La suma 0xFFFFFFF000000000 ≈ 2^{64} y el elemento 0x80000000 ≈ 2^{31}.

3-.Si nos proponemos obtener solo 1 acarreo con una lista de 32 elementos positivos iguales, se podría pensar que el objetivo es que la suma alcance 2^{31} (que ya no cabe en 32 bits como número positivo en complemento a dos). Aparentemente, cada elemento debe valer por tanto $2^{31}/32 = 2^{?}/2^{?}$ = ?. ¿Como se escribe ese valor en hexadecimal? Inicializar los 32 elementos de la lista con ese valor y comprobar si se produce el acarreo.

$2^{31}/32$ = 0x04000000

```
juanka1995@juanka-laptop ~/practicas/ec/5.2 $ ./suma_con_signo
suma = 80000000
```



En la imagen anterior vemos que el acarreo se produce debido a que los primeros 32 bits del número toman el valor 0x80000000 y la segunda parte de 32 bits esta a 0. De todas formas podriamos comprobarlo llendo a la ultima interación del bucle y viendo el flag de acarreo CF.

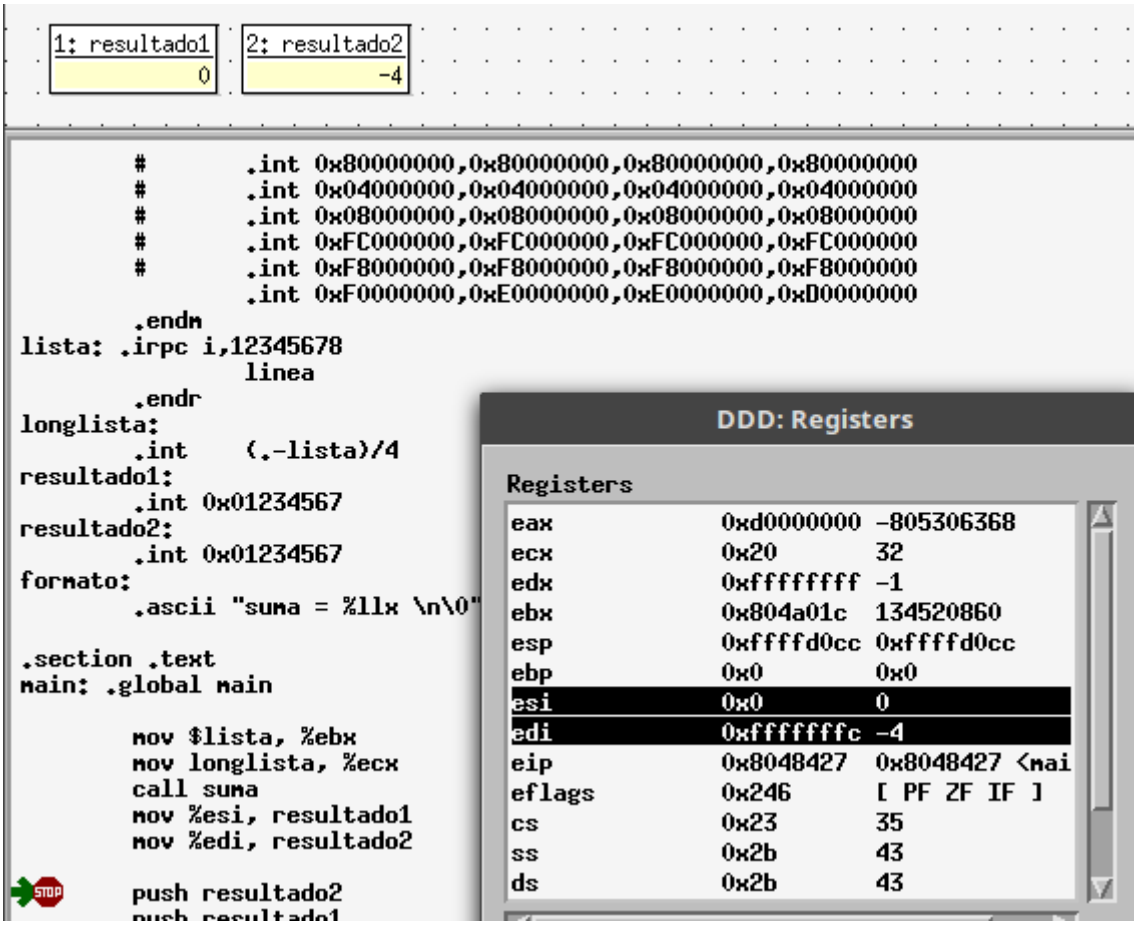
5-.Respecto a negativos, -2^{31} sí cabe en 32 bits como numero negativo en complemento a dos. Calcular que valor de elemento se requiere para obtener como suma -2^{31}, y para obtener -2^{32}. Comprobarlo usando ddd

-2^{31} = 0x80000000 = 0x04000000 * 32

-2^{32} = 0x100000000 = 0x08000000 * 32

6-.Por probar valores intermedios: si la lista se inicializara con los valores 0xF0000000, 0xE0000000, 0xE0000000, 0xD0000000, repetidos ciclicamente, ¿Que valor tomaría la suma de los 32 elementos (en hexadecimal)? Comprobarlo con ddd.

```
juanka1995@juanka-laptop ~/practicas/ec/5.2 $ ./suma_con_signo
suma = fffffffc00000000
```



El valor que tomaría la suma es de 0xFFFFFFFC00000000.