

Sesión de media.s

1. Rellenando la lista al valor -1, la media es -1. Cambiando un elemento a 0, la media pasa a valer 0. ¿Por qué? Consultar el manual de Intel sobre la instrucción de división. ¿Cuánto vale el resto o de la división en a ambos casos? Probarlo con ddd.

```
juanka1995@juanka-laptop ~/practicass/ec/5.3 $ ./media
media = -1 ; resto = 0
```

```
juanka1995@juanka-laptop ~/practicass/ec/5.3 $ ./media
media = 0 ; resto = -31
```

Esto se debe a que si cambiamos uno de los elementos a 0 el total de la suma sería -31, que dividido entre 32 daría de cociente = 0 y de resto = -31.

2. También se obtiene cociente 0 si se cambia lista[0]=1, o si lista[0]=2, ó si lista[0]=3... Comprobarlo con ddd. La siguiente pregunta lógica es hasta cuánto se puede incrementar lista[0] sin que cambie cociente=0.

Para facilitar el cálculo mental, podemos ajustar lista[1]=-2, y así la suma de todo el array vale lista[0]-32, resultando más fácil calcular el resto. ¿Para qué rango de valores de lista[0] se obtiene cociente 0? ¿Cuánto vale el resto a lo largo de ese rango? Comprobar que coinciden los signos del dividendo (suma) y del resto.

NOTA: Para evitar el ciclo editar-ensamblar-enlazar-depurar, se pueden poner un par de breakpoints antes y después de llamar la subrutina que calcula la media. Tras encontrar el primer breakpoint se puede modificar lista[0] con el comando set var lista a=<valor>. Pulsar Cont para llegar al segundo o breakpoint y ver en EAX y EDX los resultados retornados por la subrutina (que no debe hacer PUSH/POP EDX ya que no se pretende conservar el valor de EDX sino retornar el resto). Para hacer muchas ejecuciones seguidas, puede merecer la pena (re)utilizar la línea de comandos (run/set var.../cont) en lugar del ratón.

El rango de valores para lista[0] teniendo en cuenta que lista[1] = -2, es del 1 a 63, y durante ese rango el valor del cociente siempre es 0 y el del resto es el valor de la suma en cada caso(-31, -30, -29, ... , 29, 30, 31).

```
juanka1995@juanka-laptop ~/practicass/ec/5.3 $ ./media
media = -1 ; resto = 0
juanka1995@juanka-laptop ~/practicass/ec/5.3 $ gcc -m32 -g media.s -o media
juanka1995@juanka-laptop ~/practicass/ec/5.3 $ ./media
media = 0 ; resto = -31
juanka1995@juanka-laptop ~/practicass/ec/5.3 $ gcc -m32 -g media.s -o media
juanka1995@juanka-laptop ~/practicass/ec/5.3 $ ./media
media = 0 ; resto = 0
juanka1995@juanka-laptop ~/practicass/ec/5.3 $ gcc -m32 -g media.s -o media
juanka1995@juanka-laptop ~/practicass/ec/5.3 $ ./media
media = 0 ; resto = 31
juanka1995@juanka-laptop ~/practicass/ec/5.3 $ gcc -m32 -g media.s -o media
juanka1995@juanka-laptop ~/practicass/ec/5.3 $ ./media
media = 1 ; resto = 0
```

En la captura anterior hemos probado los casos en que lista[0] vale {0,1,32,63,64} en ese orden.

3. ¿Para qué rango de valores de lista[0] se obtiene media 1? ¿Cuánto vale el resto en ese rango? Comprobarlo con ddd, y notar que tanto los dividendos como los restos son positivos (el cociente se redondea hacia cero).

Teniendo en cuenta lo anterior. El rango de valores de lista[0] para que la media sea 1 es del 64 al 95 y el resto en ese rango sería de 0 a 31.

```
juanka1995@juanka-laptop ~/practicass/ec/5.3 $ gcc -m32 -g media.s -o media
juanka1995@juanka-laptop ~/practicass/ec/5.3 $ ./media
media = 0 ; resto = 31
juanka1995@juanka-laptop ~/practicass/ec/5.3 $ gcc -m32 -g media.s -o media
juanka1995@juanka-laptop ~/practicass/ec/5.3 $ ./media
media = 1 ; resto = 0
juanka1995@juanka-laptop ~/practicass/ec/5.3 $ gcc -m32 -g media.s -o media
juanka1995@juanka-laptop ~/practicass/ec/5.3 $ ./media
media = 1 ; resto = 18
juanka1995@juanka-laptop ~/practicass/ec/5.3 $ gcc -m32 -g media.s -o media
juanka1995@juanka-laptop ~/practicass/ec/5.3 $ ./media
media = 1 ; resto = 31
juanka1995@juanka-laptop ~/practicass/ec/5.3 $ gcc -m32 -g media.s -o media
juanka1995@juanka-laptop ~/practicass/ec/5.3 $ ./media
media = 2 ; resto = 0
```