Restamos R0 dos veces, si R0 es 1, no podrá restar la segunda vez y termina.

L0 R- 0 -> L1 L2

L1 R- 0 -> L3 L2

L2 R+ 0 -> L25

Si no es 1, sumamos las dos restas que habíamos hecho y continuamos con el programa.

L3 R+ 0 -> L4

L4 R+ 0 -> L5

En los siguientes bloques, se decrementa el valor de R0, cuando llega a 0 y no pueda seguir, dependiendo de cuantas veces pudo restar 1, pasará a ser par o impar:

Restamos R0 e incrementamos los contadores de los registros para los casos par e impar, así el registro que guarda el contador para el caso que R0 sea par, guardará la mitad del registro en R0, que nos servirá después para guardarlo en R0.

L5 R- 0 -> L6 L14

L6 R+ 1 -> L7

L7 R+ 2 -> L8

L8 R+ 3 -> L9

L9 R+ 4 -> L10

Restamos R0 y aumentamos los contadores de los registros para los casos impar, ya que en el caso de que R0 sea impar, este debe ser multiplicado por 3, es decir, sumado el mismo valor tres veces.

L10 R- 0 -> L11 L19

L11 R+ 2 -> L12

L12 R+ 3 -> L13

L13 R+ 4 -> L5

Si es par, por cada resta en el registro par, se suma uno a R0, luego reseteamos los demás contadores, y vuelve a la instrucción L0 para evaluar si R0 es 1:

L14 R- 2 -> L14 L15

L15 R- 3 -> L15 L16

L16 R- 4 -> L16 L17

L17 R- 1 -> L18 L0

L18 R+ 0 -> L17

Si es impar, los tres registros que guardaron el valor de R0 al principio, irán disminuyendo mientras R0 aumenta su valor, hasta quedar en 3\*R0, ya que son tres registros, al final suma uno para quedar con 3\*R0 + 1 que es el caso que nos piden en impar, luego resetea el contador de par y vuelve a L0:

L19 R- 2 -> L23 L20

L20 R- 3 -> L23 L21

L21 R- 4 -> L23 L22

L22 R- 1 -> L22 L24

L23 R+ 0 -> L19

L24 R+ 0 -> L0

Termina:

L25: HALT

benjamin.jorquera@usach.cl