

1)

Foram encontradas as seguintes substituições:

- `movia` → `movhi` e `addi`;
- `subi 1` → `addi -1`;
- `add` → `mov`;
- `labels` → endereços de memória

2) Este *breakpoint* foi colocado nesta posição para validar a decisão tomada sobre o desvio. Esta validação se dá através da verificação dos valores dos registradores neste momento.

3) Como `r4` estava apontando para o endereço 504, a operação `"ldw r5, 4(r4)"` fez com que fosse carregado o valor que estava no endereço 508, que era o 4. Disso, e sabendo que `r5` é o contador que irá controlar a iteração sobre os elementos da lista, é previsto que não será feita a varredura sobre todos os elementos, somente dos 4 primeiros. Desses, [4, 5, 3, 6], o 6 é o maior número, e será o retorno do algoritmo.

4) Esta diretiva indica que os dados do programa ficarão dispostos a partir do endereço `0x500` da memória. Esta diretiva não é necessária, uma vez que a memória pode ser manipulada através dos endereços das *labels*, mas não deixa de ser interessante ter controle da onde é iniciada a memória do programa.

5) Percebi que, em `"ldw r5, 4(r4)"` é carregado o valor 6, ainda que esteja armazenado o 7. Olhei na memória e, de fato, o número 6 está no endereço `0x504`.