

08. Desenvolver um programa monolítico, utilizando instruções rotuladas, sobre a máquina 2_REG, que implemente a função $B = A - 2$ caso o valor de A seja par, ou $B = A + 2$ caso o valor de A seja ímpar. Apresentar também a computação e a função computada para as entradas 5 e 6.

```
R1: Se a_zero então vá_para Rx senão vá_para R2;  
R2: Faça subtrair_a vá_para R3;  
R3: Se a_zero então vá_para R4 senão vá_para R7;  
R4: Faça adicionar_b vá_para R5;  
R5: Faça adicionar_b vá_para R6;  
R6: Faça adicionar_b vá_para R1;  
R7: Faça subtrair_a vá_para R8;  
R8: Se a_zero então vá_para Rx senão vá_para R5;
```

(R1, (5, 0))	(R1, (6, 0))
(R2, (5, 0))	(R2, (6, 0))
(R3, (4, 0))	(R3, (5, 0))
(R7, (4, 0))	(R7, (5, 0))
(R8, (3, 0))	(R8, (4, 0))
(R5, (3, 0))	(R5, (4, 0))
(R6, (3, 1))	(R6, (4, 1))
(R1, (3, 2))	(R1, (4, 2))
(R2, (3, 2))	(R2, (4, 2))
(R3, (2, 2))	(R3, (3, 2))
(R7, (2, 2))	(R7, (3, 2))
(R8, (1, 2))	(R8, (2, 2))
(R5, (1, 2))	(R5, (2, 2))
(R6, (1, 3))	(R6, (2, 3))
(R1, (1, 4))	(R1, (2, 4))
(R2, (1, 4))	(R2, (2, 4))
(R3, (0, 4))	(R3, (1, 4))
(R4, (0, 4))	(R7, (1, 4))
(R5, (0, 5))	(R8, (0, 4))
(R6, (0, 6))	(Rx, (0, 4))
(R1, (0, 7))	
(Rx, (0, 7))	

<TCO_08, 2_REG> : 5 -> 7

<TCO_08, 2_REG> : 6 -> 4