

26. Desenvolver um programa monolítico, utilizando instruções rotuladas, sobre a máquina 2_REG, que implemente a função $B = (A / 4) - (A \% 2)$. Apresentar a computação e a função computada para as entradas 9 e 10.

```
R1: Faça subtrair_a vá_para R2;  
R2: Faça subtrair_a vá_para R3;  
R3: Faça subtrair_a vá_para R4;  
R4: Se a_zero então vá_para Rx senão vá_para R5;  
R5: Faça subtrair_a vá_para R6;  
R6: Se a_zero então vá_para R14 senão vá_para R7;  
R7: Faça subtrair_a vá_para R8;  
R8: Se a_zero então vá_para Rx senão vá_para R9;  
R9: Faça subtrair_a vá_para R10;  
R10: Se a_zero então vá_para R14 senão vá_para R11;  
R11: Faça subtrair_a vá_para R12;  
R12: Se a_zero então vá_para Rx senão vá_para R13;  
R13: Faça adicionar_b vá_para R5;  
R14: Faça adicionar_b vá_para Rx;
```

```
(R1 , (9, 0))  
(R2 , (8, 0))  
(R3 , (7, 0))  
(R4 , (6, 0))  
(R5 , (6, 0))  
(R6 , (5, 0))  
(R7 , (5, 0))  
(R8 , (4, 0))  
(R9 , (4, 0))  
(R10, (3, 0))  
(R11, (3, 0))  
(R12, (2, 0))  
(R13, (2, 0))  
(R5 , (2, 1))  
(R6 , (1, 1))  
(R7 , (1, 1))  
(R8 , (0, 1))  
(Rx , (0, 1))
```

<PROG, 2_REG> : 9 -> 1

```
(R1 , (10, 0))  
(R2 , (9 , 0))  
(R3 , (8 , 0))  
(R4 , (7 , 0))  
(R5 , (7 , 0))  
(R6 , (6 , 0))  
(R7 , (6 , 0))  
(R8 , (5 , 0))  
(R9 , (5 , 0))  
(R10, (4 , 0))  
(R11, (4 , 0))  
(R12, (3 , 0))  
(R13, (3 , 0))  
(R5 , (3 , 1))  
(R6 , (2 , 1))  
(R7 , (2 , 1))  
(R8 , (1 , 1))  
(R9 , (1 , 1))  
(R10, (0 , 1))  
(R14, (0 , 1))  
(Rx , (0 , 2))
```

<PROG, 2_REG> : 10 -> 2