

17. Desenvolver um programa monolítico, utilizando instruções rotuladas, sobre a máquina 2_REG, que implemente a função $B = (A / 3) + (A * 2)$. Apresentar a computação e a função computada para as entradas 5 e 6.

```
R01: Se a_zero então vá_para Rx senão vá_para R02;
R02: Faça subtrair_a vá_para R03;
R03: Faça adicionar_b vá_para R04;
R04: Faça adicionar_b vá_para R05;
R05: Se a_zero então vá_para Rx senão vá_para R06;
R06: Faça subtrair_a vá_para R07;
R07: Faça adicionar_b vá_para R08;
R08: Faça adicionar_b vá_para R09;
R09: Se a_zero então vá_para Rx senão vá_para R10;
R10: Faça subtrair_a vá_para R11;
R11: Faça adicionar_b vá_para R12;
R12: Faça adicionar_b vá_para R13;
R13: Faça adicionar_b vá_para R01;
```

(R01, (5, 0))	(R01, (6, 0))
(R02, (5, 0))	(R02, (6, 0))
(R03, (4, 0))	(R03, (5, 0))
(R04, (4, 1))	(R04, (5, 1))
(R05, (4, 2))	(R05, (5, 2))
(R06, (4, 2))	(R06, (5, 2))
(R07, (3, 2))	(R07, (4, 2))
(R08, (3, 3))	(R08, (4, 3))
(R09, (3, 4))	(R09, (4, 4))
(R10, (3, 4))	(R10, (4, 4))
(R11, (2, 4))	(R11, (3, 4))
(R12, (2, 5))	(R12, (3, 5))
(R13, (2, 6))	(R13, (3, 6))
(R01, (2, 7))	(R01, (3, 7))
(R02, (2, 7))	(R02, (3, 7))
(R03, (1, 7))	(R03, (2, 7))
(R04, (1, 8))	(R04, (2, 8))
(R05, (1, 9))	(R05, (2, 9))
(R06, (1, 9))	(R06, (2, 9))
(R07, (0, 9))	(R07, (1, 9))
(R08, (0, 10))	(R08, (1, 10))
(R09, (0, 11))	(R09, (1, 11))
(Rx, (0, 11))	(R10, (1, 11))
	(R11, (0, 11))
	(R12, (0, 12))
	(R13, (0, 13))
	(R01, (0, 14))
	(RX, (0, 14))

<TCO_17, 2_REG> : 5 -> 11

<TCO_17, 2_REG> : 6 -> 14