**Código intermediário → Código executável**

**Máquina de Pilha**

* **Código que deve ser gerado**

**Instruções de manipulação de dados**

* + **PUSH v – Empilha a constante v no topo da pilha**
    - **PUSH CONST NUM**
  + **POP - Desempilha o valor no topo**
    - **Não retorna valor**
  + **Ex:**

**PUSH 5 | 5 | | 8 |**

**PUSH 3 | 3 |**

**ADD**

* **COPY - Duplicar o valor do topo**
* **STOR x - Desempilha e armazena**
* **EX:**

**LOAD B**

**LOAD C**

**ADD**

**STOR A**

**Instruções aritméticas inteiras**

* **ADD**
* **SUB ( sub\_top - topo)**
* **MUL**
* **DIV**

**Instruções aritméticas de ponto flutuante**

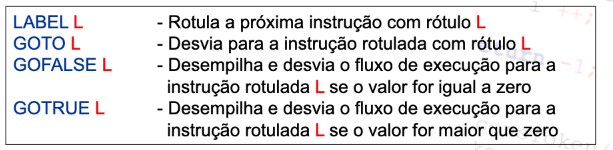
* **ADD f**
* **SUB f ( sub\_top - topo)**
* **MUL**
* **DIV**

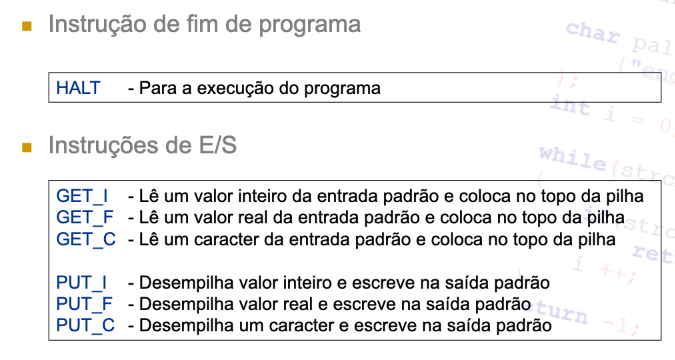
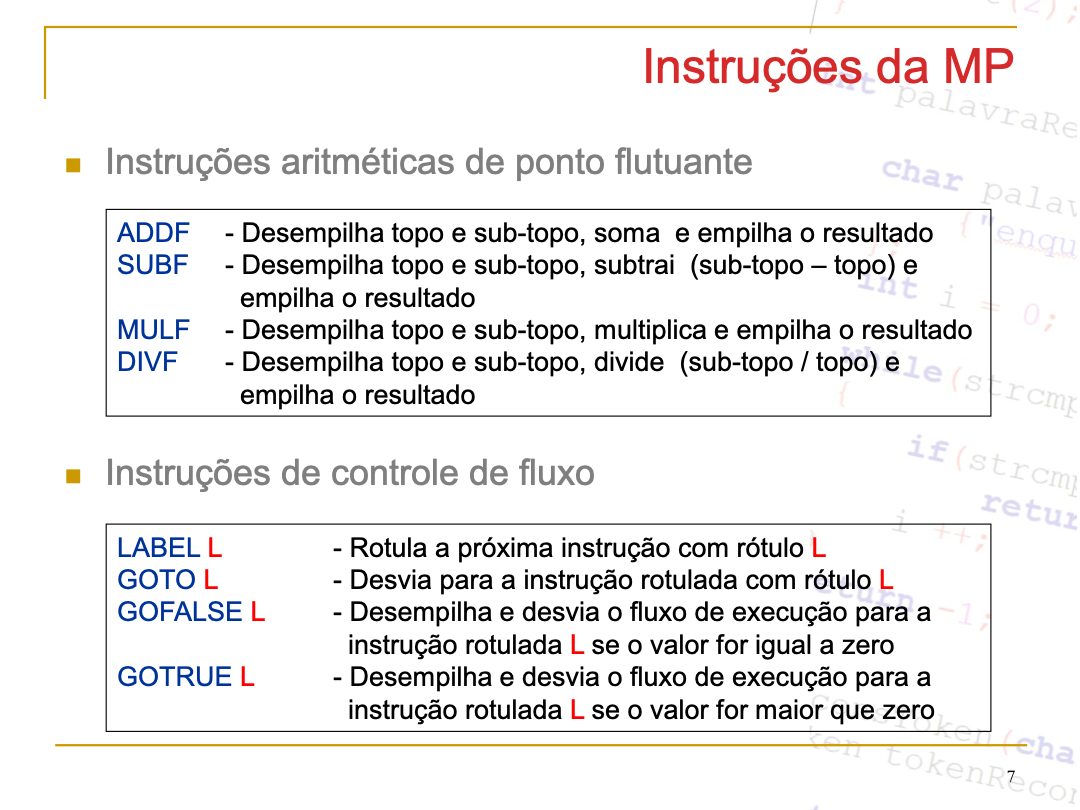
**Instruções de controle de fluxo**

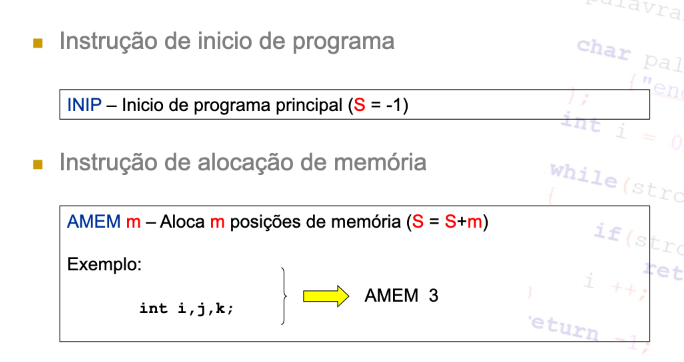
* **LABEL L - Marcador de desvio**
  + **Ex**

**LABEL L1**

**LOAD B **

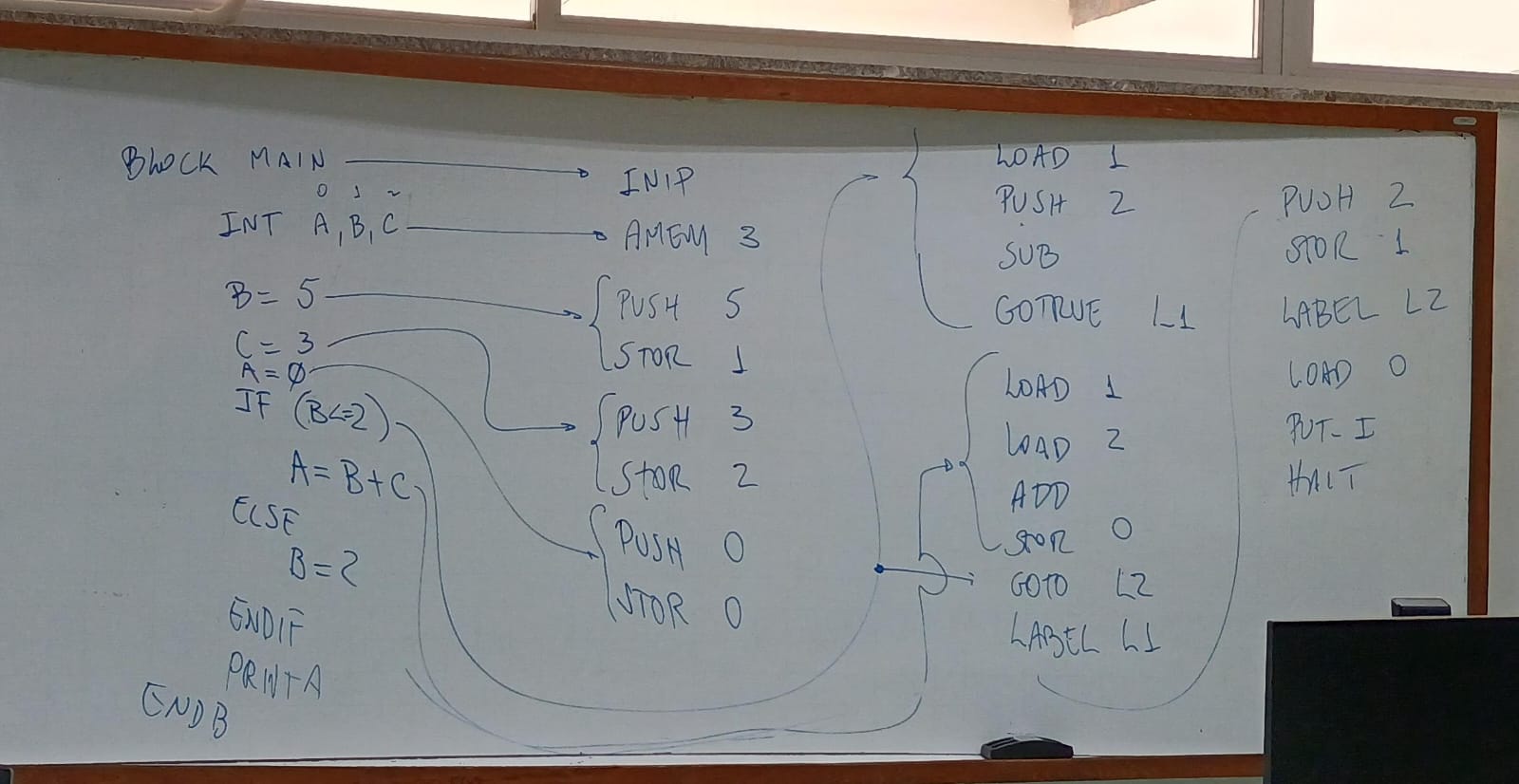
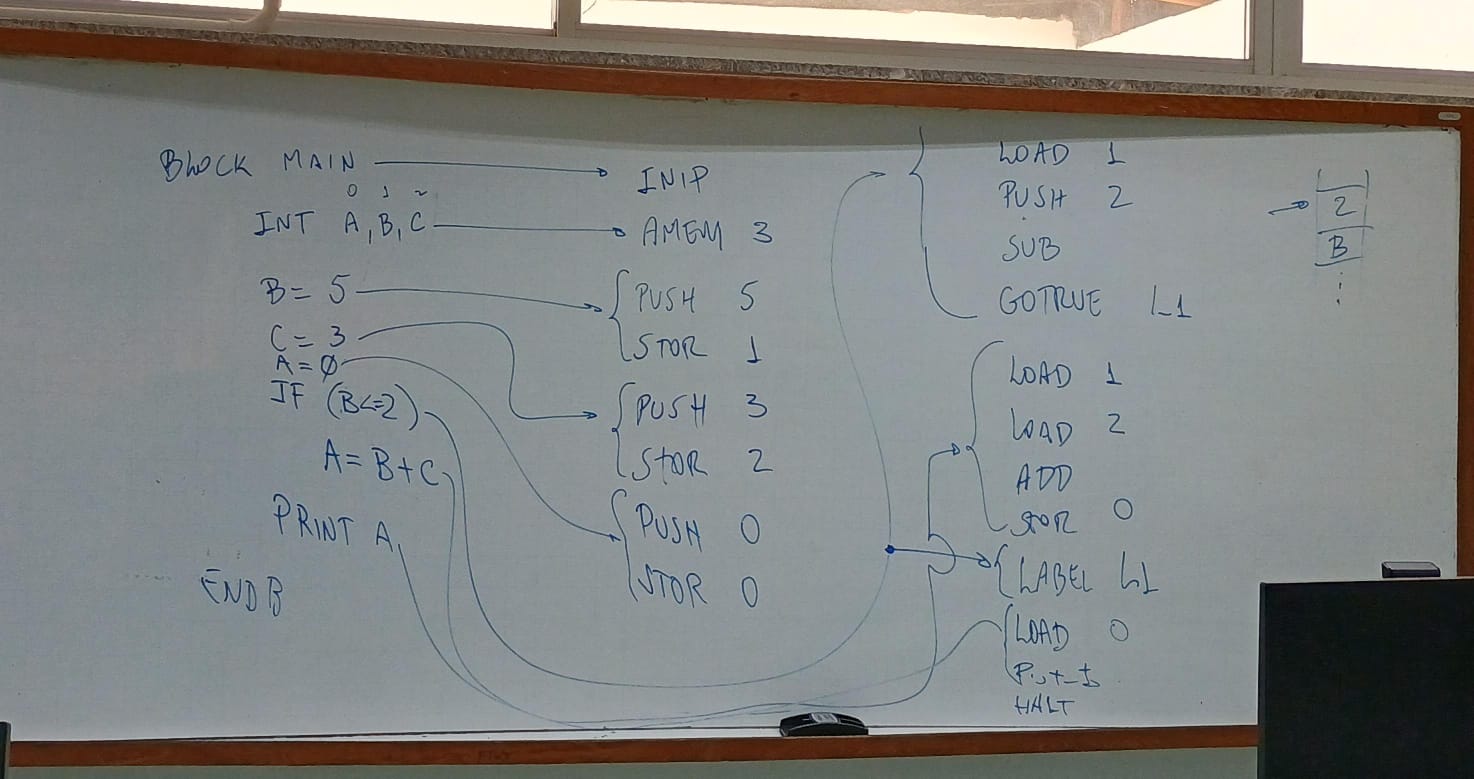
****

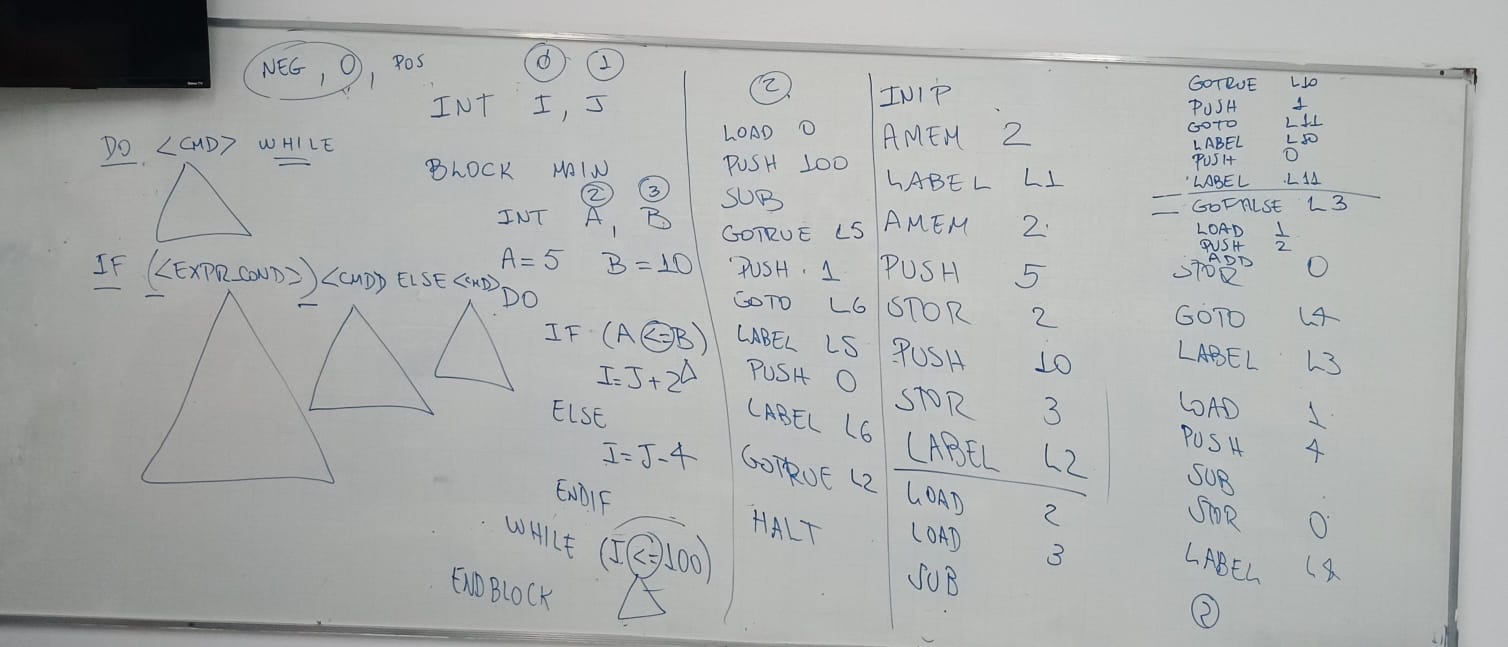
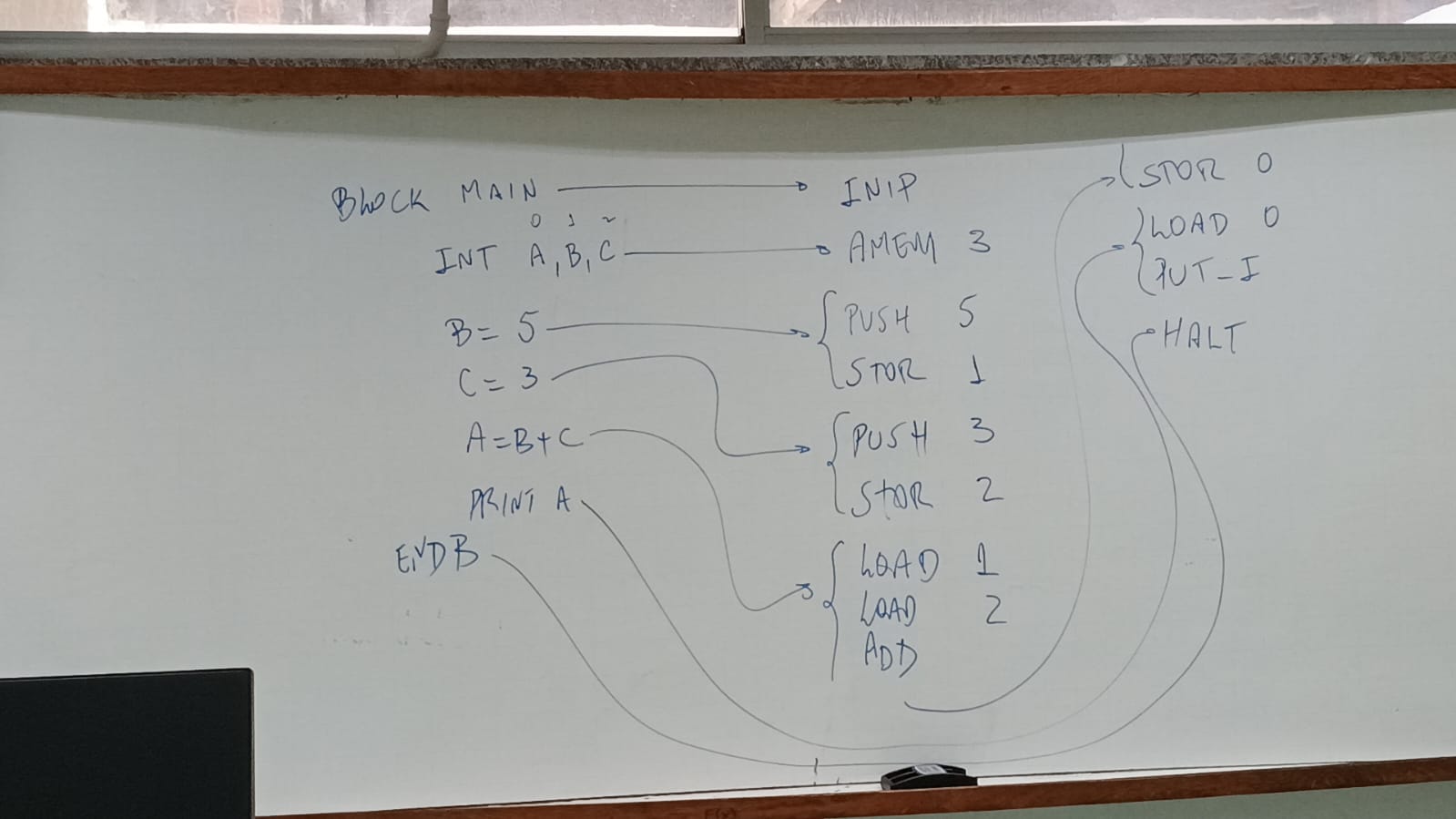
****

****

**Exemplos de programa**

* **Comparação é sempre um subtração**

****

****

**→ Esquemas de tradução: “ Deixar zero ou um na pilha. ”**

**0 → FALSE**

**1 → TRUE**

**GOFALSE L : Se valor de L for igual a 0**

**GOTRUE L: Se valor de L for maior que 0**

**Ex:**

**while <exp\_cond> do**

**<bloco\_cmd>**

**endwhile**

**Código da MP**

**LABEL Ly**

**→ Comandos da expressão**

**GOFALSE Lx**

**→ Comandos do bloco de comando**

**GOTO Ly**

**LABEL Lx**

**Ex:**

**if <exp\_cond> then**

**<bloco\_cmd>**

**endif**

**Código da MP**

**// Como não é loop não precisa de label**

**→ Comandos da expressão**

**GOFALSE Lx**

**→ Comandos do bloco de comando**

**GOTO Lx**

**LABEL Lx**

**Ex:**

**if <exp\_cond> then**

**<bloco\_cmd>**

**else**

**<bloco\_cmd>**

**endif**

**Código da MP**

**// Como não é loop não precisa de label**

**→ Comandos da expressão**

**GOFALSE Lx**

**→ Comandos do bloco de comando**

**GOTO Ly**

**LABEL Lx**

**→ Comandos do bloco de comando**

**LABEL Ly**

**Ex: A > B**

**Código da MP**

**LOAD A**

**LOAD B**

**SUB**

**GOTRUE Lx**

**PUSH 0**

**GOTO Ly**

**LABEL Lx**

**PUSH 1**

**LABELLy**

**Ex: A >= B**

**Código da MP**

**LOAD A**

**LOAD B**

**SUB**

**COPY**

**GOTRUE Lx**

**GOFALSE Ly**

**PUSH 0**

**GOTO Lz**

**POP**

**LABEL Ly**

**PUSH 1**

**LABEL Lz**

**Ex: A == B**

**Código da MP**

**LOAD A**

**LOAD B**

**SUB**

**GOFALSE Ly**

**PUSH 0**

**GOTO Lx**

**LABEL L y**

**PUSH 1**

**LABEL Lx**

**Ex: A != B**

**LOAD A**

**LOAD B**

**SUB**

**COPY**

**GOFALSE Lx**

**POP**

**PUSH 1**

**LABEL Lx**

**Ex: A <= B**

**Código da MP**

**LOAD A**

**LOAD B**

**SUB**

**GOTRUE Lx**

**PUSH 1**

**GOTO Ly**

**LABEL Lz**

**PUSH 0**

**LABEL Ly**

**Ex: A < B**

**Código da MP**

**LOAD A**

**LOAD B**

**SUB**

**COPY**

**GOFALSE Lx**

**GOTRUE Ly**

**PUSH 1**

**GOTO Lz**

**LABEL Lx**

**POP**

**LABEL Ly**

**PUSH 0**

**LABEL Lz**