

CESAR

(As bases da civilização atual)

Pilhas



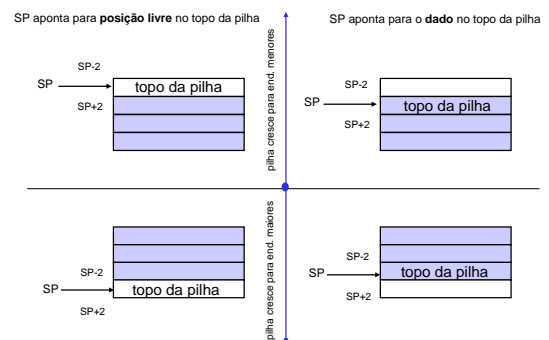
2

Pilha

- variações
 - pilha crescendo para os endereços maiores de memória
 - pilha crescendo para os endereços menores
 - SP aponta para o dado no topo da pilha
 - SP aponta para posição livre no topo da pilha

3

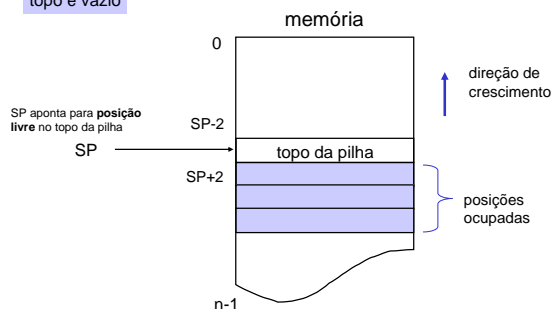
4 possíveis implementações



4

Pilha cresce para end. menores

topo é vazio



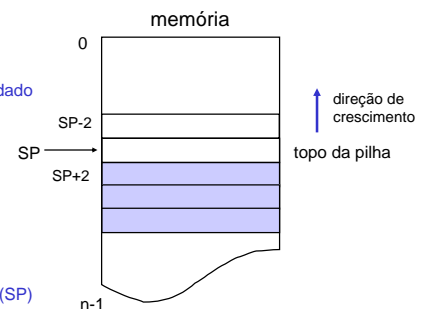
5

Pilha cresce para end. menores

topo é vazio

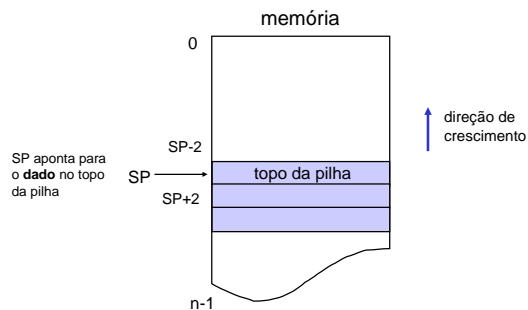
Inserção
 $\text{Mem}(\text{SP}) \leftarrow \text{dado}$
 $\text{SP} \leftarrow \text{SP} - 2$

Retirada
 $\text{SP} \leftarrow \text{SP} + 2$
 $\text{dado} \leftarrow \text{Mem}(\text{SP})$



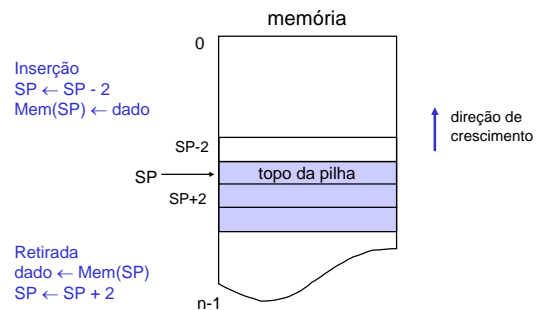
6

Pilha cresce para end. menores



7

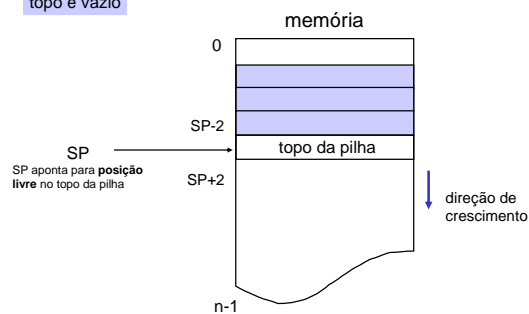
Pilha cresce para end. menores



8

Pilha cresce para end. maiores

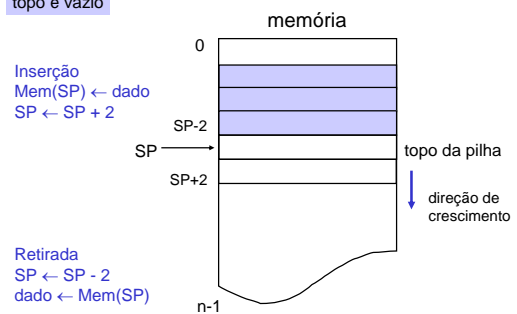
topo é vazio



9

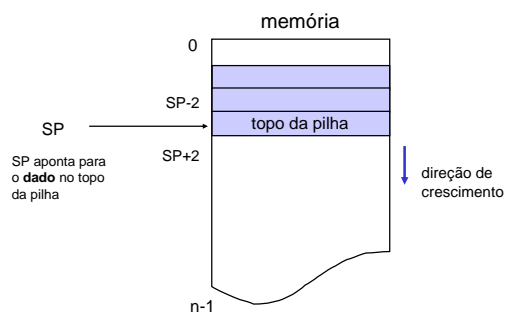
Pilha cresce para end. maiores

topo é vazio



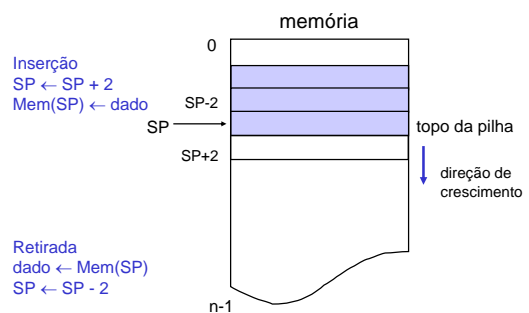
10

Pilha cresce para end. maiores



11

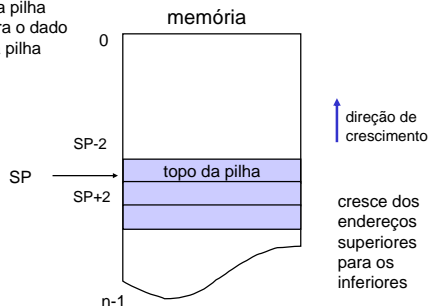
Pilha cresce para end. maiores



12

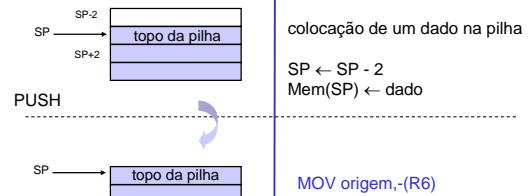
Pilha no Cesar

ponteiro da pilha aponta para o dado no topo da pilha



13

Pilha no Cesar: push



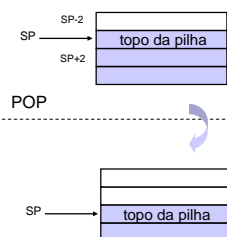
14

Pilha no Cesar: pop

retirada de um dado da pilha

dado ← Mem(SP)
SP ← SP + 2

MOV (R6)+,destino



15

Endereçamento com SP (R6)

R6 no campo de destino

MOV R0, R6	R6 ← R0	
MOV R0, (R6)+	Mem(R6) ← R0; R6 ← R6 + 2	
MOV R0, -(R6)	R6 ← R6 - 2; Mem(R6) ← R0	PUSH
MOV R0, ddd(R6)	Mem(R6+ddd) ← R0	
MOV R0, (R6)	Mem(R6) ← R0	
MOV R0, ((R6)+)	Mem(Mem(R6)) ← R0; R6 ← R6 + 2	
MOV R0, -(R6))	R6 ← R6 - 2; Mem(Mem(R6)) ← R0	
MOV R0, (ddd(R6))	Mem(Mem(R6+ddd)) ← R0	

16

Endereçamento com SP (R6)

R6 no campo de fonte

→ MOV (R6)+, R0	R0 ← Mem(R6); R6 ← R6 + 2	POP
→ MOV -(R6), R0	R6 ← R6 - 2; R0 ← Mem(R6)	
→ MOV (R6), R0	R0 ← Mem(R6)	copiar sem remover

17

Endereçamento com SP (R6)

Útil Cuidado ! Não usar !!

MOV R0, R6	R6 ← R0	
MOV R0, (R6)+	Mem(R6) ← R0; R6 ← R6 + 2	
→ MOV R0, -(R6)	R6 ← R6 - 2; Mem(R6) ← R0	; (PUSH)
MOV R0, ddd(R6)	Mem(R6+ddd) ← R0	
MOV R0, (R6)	Mem(R6) ← R0	
MOV R0, ((R6)+)	Mem(Mem(R6)) ← R0; R6 ← R6 + 2	
→ MOV R0, -(R6))	R6 ← R6 - 2; Mem(Mem(R6)) ← R0	
MOV R0, (ddd(R6))	Mem(Mem(R6+ddd)) ← R0	
→ MOV (R6)+, R0	R0 ← Mem(R6); R6 ← R6 + 2	; (POP)
→ MOV -(R6), R0	R6 ← R6 - 2; R0 ← Mem(R6)	
→ MOV (R6), R0	R0 ← Mem(R6)	; (COPIAR SEM REMOVER)

18