

## Compilation via la sémantique

(simplifiée)

On doit **séparer** les structures de données que doit manipuler le compilateur de celles utilisées à l'exécution du programme:

- La mémoire  $\mu$  ne sert qu'à l'**exécution**.
- L'environnement  $\rho$  sert:

- au **compilateur**:

$$\hat{\rho} : Var \rightarrow Code$$

dit au compilateur quel bout de code produire pour retrouver la valeur de chaque variable et l'empiler.

- à l'**exécution**:

Dans les clôtures, on réservera un tableau  $R$  stockant les valeurs des variables locales

## Jugements de compilation

---

Jugement  $\hat{\rho} \vdash_c M \Rightarrow Code$

“compiler  $M$  en les instructions  $Code$ ”

Note: on va utiliser la pile pour retourner la valeur de  $M$ .

$$\frac{}{\hat{\rho} \vdash_c x \Rightarrow \hat{\rho}(x)} (Var_c)$$

## Application de fonction

---

$\hat{\rho} \vdash_c M \Rightarrow Code_1$  ... à l'exécution, on aura l'adresse

d'une clôture ( $adr.de\ Code_3, V_1, \dots, V_n$ ) au sommet de la pile

$\hat{\rho} \vdash_c N \Rightarrow Code_2$  ... et maintenant

$adr.de\ (adr.de\ Code_3, V_1, \dots, V_n)$  [en  $4(\%esp)$ ] et  $V$  [en  $(\%esp)$ ]

```
CallCode = mov 4(%esp), %eax
           mov (%eax), %eax
           call *%eax
           add $8, %esp
           push %eax
```

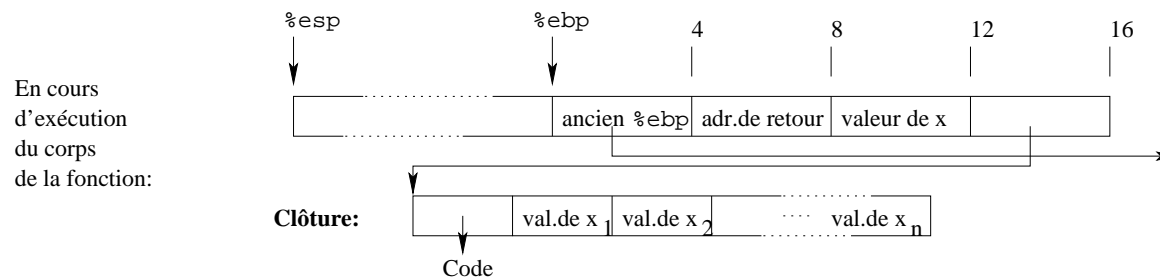
**Exercice:** dessiner  
la pile au cours de *CallCode*  
(on suppose que *Code<sub>3</sub>*  
retourne son résultat  
dans *%eax*)

---

$\hat{\rho} \vdash_c MN \Rightarrow Code_1.Code_2.CallCode$

# Fonctions

Soient  $x_1, \dots, x_n$  les variables **libres** dans  $M$ , sauf  $x$ .



$$\hat{\rho}' = \{x_i \mapsto \text{push } 4i(\%edi) \mid i = 1..n\} \cup \{x \mapsto \text{push } 8(\%ebp)\}$$

$$Tete = \text{push } \%ebp; \text{mov } \%esp, \%ebp; \text{push } \%edi; \text{mov } 12(\%ebp), \%edi$$

$$Fin = \text{pop } \%eax; \text{mov } -4(\%ebp), \%edi; \text{mov } \%ebp, \%esp; \text{pop } \%ebp; \text{ret}$$

$$\hat{\rho}' \vdash_c M \Rightarrow Corps$$

$$Code = \text{push } \$4(n+1); \text{call } \text{malloc}; \text{mov } \%eax, (\%esp)$$

$$\text{mov } \$(Tete; Corps; Fin), (\%eax); [\hat{\rho}(x_i); \text{pop } 4i(\%eax) \mid i = 1..n]$$

$$\hat{\rho} \mapsto \text{fun } x \rightarrow M \Rightarrow Code$$