

### SUBIECTUL 3:

#### Funcția universală:

Fie  $P$  un program standard,  $P: \bar{I}_1, \dots, \bar{I}_m$ .

$\#P = [\#(\bar{I}_1), \#(\bar{I}_2), \dots, \#(\bar{I}_m)] - 1$ , unde:  $\#(\bar{I}) = \langle a, \langle b, c \rangle \rangle$

nr. atare  
variabile

eticheta

tipul  
inote

$f_P^{(n)}(x_1, \dots, x_m)$  - funcția calculată de  $P$ .

$\phi^{(m+1)}(x_1, \dots, x_m, t) = f_P^{(n)}(x_1, \dots, x_m)$  cu  $\#(P) = t$

primește  
 $m+1$  var.  
la intrare

"  
funcția universală de  $n$  variabile

TEOREMA 1: Pentru orice  $n \geq 1$ , funcția  $\phi^{(m+1)}$  este calculabilă  
cu programe standard.

→ Dem:  $T \leftarrow X_{m+1} + 1$  // numărul codificării lui  $P$

$S = \bigcup_{i=1}^m P_{2i}^{x_i}$  // notarea programului

$k \leftarrow 1$  // contor pt. instrucțiunea curentă

if  $(k > L_T(T)) \vee (k = 0)$  GO TO  $L_1$

$\bar{I} = \pi(\pi(T)_k)$  // tipul instrucțiunii

$V \leftarrow \ell(\pi(T)_k)$  // numărul asociat instrucțiunii

if  $(\bar{I} = 0)$  GO TO  $N$  // efect nul

if  $(\bar{I} = 1)$  GO TO  $A$  // incrementare

if  $(\bar{I} = 2)$  GO TO  $M$  // decrementare



if ( $P_v \neq S$ ) GO TO N

$k \leftarrow \min_R [l(T)_k] = j-2$  &  $(j-2) > 1$  // m-<sup>al</sup> do stădine al primei  
instrucțiuni care are eticheta  
 $j-2$ , dacă există

GO TO C

A:  $S \leftarrow S * P_v$

GO TO N

N: if ( $P_v \neq S$ ) GO TO N

$S \leftarrow S / P_v$

N:  $k \leftarrow k + 1$

GO TO C

L1:  $y \leftarrow (S)_1$

\* TEOREMA 2: Funcția HALT nu este calculabilă cu programe standard.

↳ Dem: Presupunem prin reducere la absurd că HALT este calculabilă cu programe standard.

Considerăm programul P a.7.  $\varphi_p^{(2)}(x_1, x_2) = \text{HALT}$ .

A: if HALT(x, x) GO TO A  $\rightsquigarrow$  P',  $\#(P') = 2$

$\text{HALT}(x, x) = \neg \text{HALT}(x, x), \neq x$

$x = x$

$\Downarrow$

$\text{HALT}(x, x) = \neg \text{HALT}(x, x) \quad \times$



\* Definim  $\text{HALT}(x, y) = \begin{cases} 1, & \text{dacă programul } P \text{ cu } \#(P) = y \text{ se termină} \\ & \text{pe intrarea } x \\ 0, & \text{altfel} \end{cases}$

### Problema oprirei

Nu există nicio mașină Turing care, cunoscând codul unei mașini  $M$  și codul lui  $w$ , să poată decide dacă mașina  $M$  se oprește pe intrarea  $w$ .