

Exercício 06

Hemílio Lauro e Matheus Barbosa

Pascal

```
function max (i, j: integer): integer;  
( return maximum of integers I and j)  
begin  
  if i > j then max := i  
  else max := j  
end;
```

TOKEN	CÓDIGO
function	simb_func
max	simb_nfunc
(simb_apar
i	id
,	simb_virg
j	id
:	simb_def
integer	id
)	simb_fpar
:	simb_def
integer	id
;	simb_pv
begin	simb_begin
i	id
>	simb_maior
j	id
then	simb_then
max	simb_func
:=	simb_atrib
i	id
else	simb_else

Exercício 06

Hemílio Lauro e Matheus Barbosa

max	simb_func
:=	simb_atrib
j	id
end	simb_end
;	simb_pv

```
C
int max (i, j) int i, j;
{ /* maximum of integers i and j */
return i > j ? i : j;
}
```

TOKEN	CÓDIGO
int	id
max	simb_func
(simb_apar
i	id
,	simb_vig
j	id
)	simb_fpar
int	id
i	id
,	simb_vig
j	id
;	simb_pv
{	simb_acol
return	simb_return
i	id
>	simb_maior
j	id

Exercício 06

Hemílio Lauro e Matheus Barbosa

?	simb_ternario
i	id
:	simb_condicao
j	id
;	simb_pv
}	simb_fcol