

Exercício Tabela Símbolos

terça-feira, 26 de novembro de 2024 11:20

Execute o "teste de mesa" do funcionamento da Tabela de Símbolos, conforme visto em sala de aula, para o seguinte programa fonte Proc:

// Ponto 0

```
int a, b
```

```
prot subtrair(&int, int, int)
```

```
def somar(&int r, int a, int b)
```

```
    int z
```

```
    z = a + b
```

```
    r = z
```

```
endp
```

//Ponto 1

```
def subtrair(&int r, int a, int b)
```

```
    int w
```

```
    w = a - b
```

```
    r = w
```

// Ponto 2

```
endp
```

```
def multiplicar(&int r, int a, int b)
```

```
    int x, y
```

```
    x = a
```

```
    y = b
```

```
    r = x * y
```

```
endp
```

```
def dividir(&int r, int a, int b)
```

```
    if (b != 0) r = a / b
```

```
    else
```

```
        putstr "Divisao por zero!"
```

```
        r = 0
```

```
    endi
```

```
endp
```

// Ponto 3

```
def init
```

```
    int z
```

```
    a = 10
```

```
    b = 5
```

// Ponto 4

```
do somar(z, a, b)
  putstr "Soma: "
  putint z
  putchar '\n'

do subtrair(z, a, b)
  putstr "Subtracao: "
  putint z
  putchar '\n'

do multiplicar(z, a, b)
  putstr "Multiplicacao: "
  putint z
  putchar '\n'

do dividir(z, a, b)
  putstr "Divisao: "
  putint z
  putchar '\n'
```

endp

Estado da Tabela de Símbolos nos diversos pontos indicados como comentário no código acima

Ponto 0:

Lexema	Escopo	Tipo	Categoria	Passagem	Zumbi	Array	Dim1	Dim2	Endereço	Rótulo	...

TOPO = -1 (sombreamento verde na TS; TS vazia; primeira linha tem índice 0)

Ponto 1:

Lexema	Escopo	Tipo	Categoria	Passagem	Zumbi	Array	Dim1	Dim2	Endereço	Rótulo	...
--------	--------	------	-----------	----------	-------	-------	------	------	----------	--------	-----

a	GBL	INT	VG	X	X	VS	0	0			
b	GBL	INT	VG	X	X	VS	0	0			
subtrair	GBL	X	PRT	X	X	X	0	0			
	LCL	INT	PAR	REF	X	VS	0	0			
	LCL	INT	PAR	VAL	X	VS	0	0			
	LCL	INT	PAR	VAL	X	VS	0	0			
somar	GBL	X	PRC	X	X	X	0	0			
r	LCL	INT	PAR	REF	ZMB	VS	0	0			
a	LCL	INT	PAR	VAL	ZMB	VS	0	0			
b	LCL	INT	PAR	VAL	ZMB	VS	0	0			
z	LCL	INT	VL	X	X	VS	0	0			

TOPO = 9 (sombreamento verde na TS; varável local "z", do procedimento somar, que ocupa a posição 10 da TS, foi excluída)

Ponto 2:

Lexema	Escopo	Tipo	Categoria	Passagem	Zumbi	Array	Dim1	Dim2	Endereço	Rótulo	...
a	GBL	INT	VG	X	X	VS	0	0			
b	GBL	INT	VG	X	X	VS	0	0			
subtrair	GBL	X	PRC	X	X	X	0	0			
r	LCL	INT	PAR	REF	VIV	VS	0	0			
a	LCL	INT	PAR	VAL	VIV	VS	0	0			
b	LCL	INT	PAR	VAL	VIV	VS	0	0			
somar	GBL	X	PRC	X	X	X	0	0			
r	LCL	INT	PAR	REF	ZMB	VS	0	0			
a	LCL	INT	PAR	VAL	ZMB	VS	0	0			
b	LCL	INT	PAR	VAL	ZMB	VS	0	0			
w	LCL	INT	VL	X	X	VS	0	0			

TOPO = 10 (sombreamento verde na TS)

Ponto 3:

Lexema	Escopo	Tipo	Categoria	Passagem	Zumbi	Array	Dim1	Dim2	Endereço	Rótulo	...
a	GBL	INT	VG	X	X	VS	0	0			
b	GBL	INT	VG	X	X	VS	0	0			
subtrair	GBL	X	PRC	X	X	X	0	0			
r	LCL	INT	PAR	REF	ZMB	VS	0	0			
a	LCL	INT	PAR	VAL	ZMB	VS	0	0			
b	LCL	INT	PAR	VAL	ZMB	VS	0	0			
somar	GBL	X	PRC	X	X	X	0	0			
r	LCL	INT	PAR	REF	ZMB	VS	0	0			

a	LCL	INT	PAR	VAL	ZMB	VS	0	0			
b	LCL	INT	PAR	VAL	ZMB	VS	0	0			
multiplicar	GBL	X	PRC	X	X	X	0	0			
r	LCL	INT	PAR	REF	ZMB	VS	0	0			
a	LCL	INT	PAR	VAL	ZMB	VS	0	0			
b	LCL	INT	PAR	VAL	ZMB	VS	0	0			
dividir	GBL	X	PRC	X	X	X	0	0			
r	LCL	INT	PAR	REF	ZMB	VS	0	0			
a	LCL	INT	PAR	VAL	ZMB	VS	0	0			
b	LCL	INT	PAR	VAL	ZMB	VS	0	0			

TOPO = 17 (sombreamento verde na TS)

Ponto 4:

Lexema	Escopo	Tipo	Categoria	Passagem	Zumbi	Array	Dim1	Dim2	Endereço	Rótulo	...
a	GBL	INT	VG	X	X	VS	0	0			
b	GBL	INT	VG	X	X	VS	0	0			
subtrair	GBL	X	PRC	X	X	X	0	0			
r	LCL	INT	PAR	REF	ZMB	VS	0	0			
a	LCL	INT	PAR	VAL	ZMB	VS	0	0			
b	LCL	INT	PAR	VAL	ZMB	VS	0	0			
somar	GBL	X	PRC	X	X	X	0	0			
r	LCL	INT	PAR	REF	ZMB	VS	0	0			
a	LCL	INT	PAR	VAL	ZMB	VS	0	0			
b	LCL	INT	PAR	VAL	ZMB	VS	0	0			
multiplicar	GBL	X	PRC	X	X	X	0	0			
r	LCL	INT	PAR	REF	ZMB	VS	0	0			
a	LCL	INT	PAR	VAL	ZMB	VS	0	0			
b	LCL	INT	PAR	VAL	ZMB	VS	0	0			
dividir	GBL	X	PRC	X	X	X	0	0			
r	LCL	INT	PAR	REF	ZMB	VS	0	0			
a	LCL	INT	PAR	VAL	ZMB	VS	0	0			
b	LCL	INT	PAR	VAL	ZMB	VS	0	0			
z	LCL	INT	VL	X	X	VS	0	0			

TOPO = 18 (sombreamento verde na TS)

