

Lista de Exercícios Geração de Código Intermediário

1) Qual o principal motivo para utilizarmos representações intermediárias na implementação de compiladores?

2) Traduza para a representação de três endereços os códigos abaixo. Neste exercício, não é necessário usar esquema de tradução.

<pre>a) // Função Fibonacci int fibonacci() { int anterior = 0, proximo = 1, aux, i, n; scanf("%d", &n) if(n == 1) printf("0"); else if(n == 2) printf("1"); else { for(i = 3; i <= n; i++) { aux = proximo; proximo = anterior + proximo; anterior = aux; } printf ("%d", proximo); } }</pre>	<pre>b) int primo() { int n, nMax, i, j, totalDivisores; scanf("%d", &nMax); for(i = 0; i <= nMax; i++) { totalDivisores = 0; for(j = 1; j <= i; j++) { if(i % j == 0) totalDivisores++; } if(totalDivisores == 2) { printf("%d", i); } } }</pre>
---	---

OBS: Usar a instrução READ <var> e PRINT <var> no código de 3 endereços no lugar de scanf e printf.
Usar MOD <var> <var> para %.

3) Usando os esquemas de tradução dos slides da aula “Geração de Código Intermediário Parte II – Traduzindo para IR” (arquivo 11-Codigo_Intermediario_pt2.pdf), faça:

- A tradução dos seguintes trechos de código em representação de 3 endereços;
- Mostre o estado da tabela de símbolos.

Para cada item, explique em detalhes quais foram os esquemas utilizados e como você realizou a tradução.

<p>a)</p> <pre>x : integer; y : integer; z : real; while y > x do z = z + x y = y - x</pre>	<p>b)</p> <pre>a : integer; b : integer; c : integer; maior : integer maior = a if (maior < b) maior = b if (maior < c) maior = c</pre>
---	--

Exercícios Dragão 2ª ed:

- 6.5.1
- 6.6.1 (usar o esquema de tradução visto em aula)
- 6.6.3