Nome: Tiago de Luna Farias
Nusp: 9875503

Prova 2 de Compiladores - Respostas

1) Determine os tipos e os endereços relativos para os identificadores na sequência de declarações a seguir. Assuma que um **int** ocupa quatro bytes e um **float** oito bytes:

a) float x

Id	Tipo	Deslocamento	Evn
x	float	0	1

b) record { float x; float y; } p;

Id	Tipo	Deslocamento	Evn
x	float	0	2
y	float	8	2
p	record()	8	1

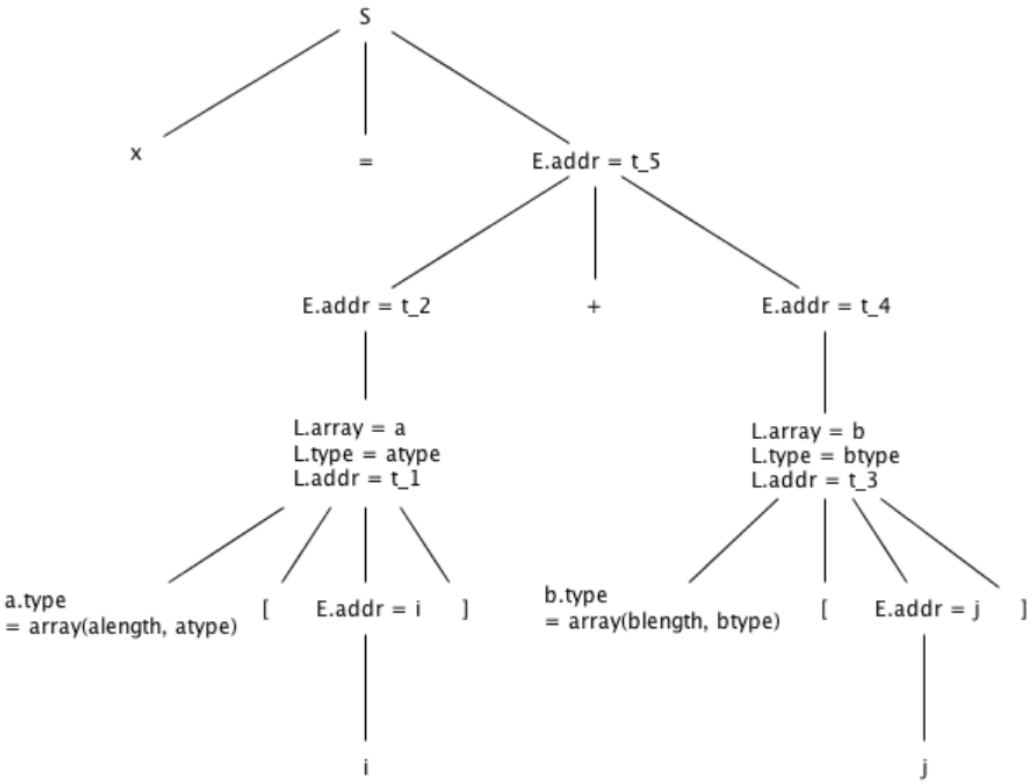
c) record { int tag; float x; float y; } q;

Id	Tipo	Deslocamento	Evn
tag	int	0	3
x	float	4	3
y	float	12	3
q	record()	24	1

2) Utilize o esquema de tradução da Fig. 6.22 (apresentada a seguir) para traduzir as seguintes atribuições:

a) x = a[i]+ b[j]

análise da árvore:



Código de 3 endereços:

t_1 = i * awidth

t_2 = a[t_1]

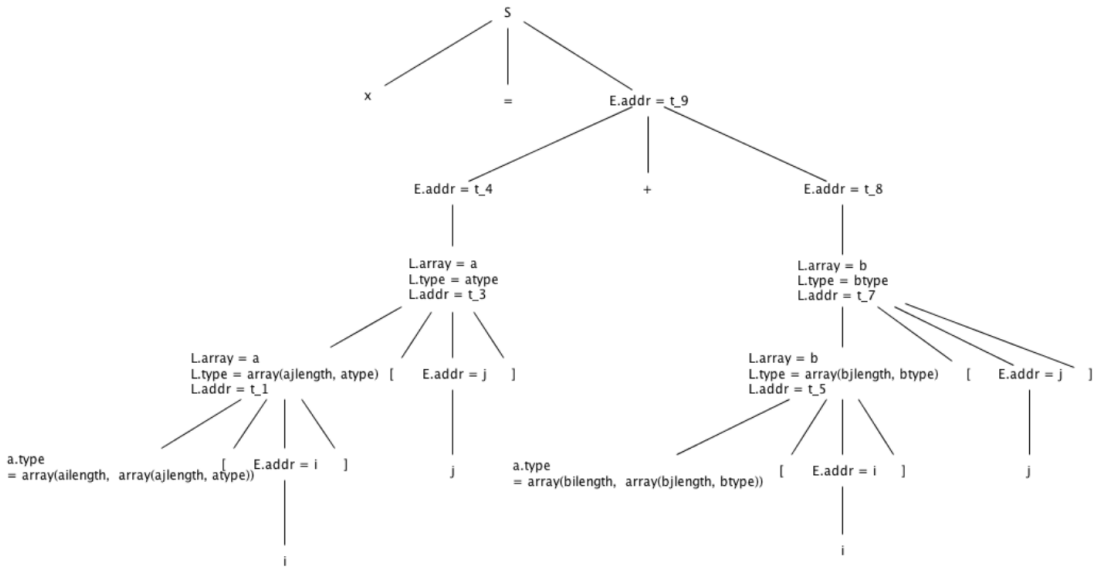
t_3 = j * bwidth

t_4 = b[t_3]

t_5 = t_2 + t_4

x = t_5

b) x = a[i][j] + b[i][j]
análise da árvore:



Código de 3 endereços:
t_1 = i * ai_width
t_2 = j * aj_width
t_3 = t_1 + t_2
t_4 = a[t_3]
t_5 = i * bi_width
t_6 = j * bj_width
t_7 = t_5 + t_6
t_8 = b[t_7]
t_9 = t_4 + t_8
x = t_9

c) x = a[b[i][j]][c[k]]

3) Um vetor de inteiros A[i,j] possui um índice i variando de 1 a 10 e um índice j variando de 1 a 20. Inteiros utilizam 4 bytes cada. Suponha que o vetor A é armazenado começando no byte 0. Encontre a localização de:

Fórmula de cálculo: ((i-1) * 20 + (j-1)) * 4

- A[4,5]

(3 * 20 + 4) * 4 = 256

- A[10,8]

(9 * 20 + 7) * 4 = 748

- A[3,17]

(2 * 20 + 16) * 4 = 224

4) Refaça a questão anterior considerando que o armazenamento do vetor A é ordenado pelas colunas (*column-major order*).

Fórmula de cálculo: ((j-1) * 10 + (j-1)) * 4

- A[4,5]

(4 * 10 + 3) * 4 = 172

- A[10,8]

(7 * 10 + 9) * 4 = 316

- A[3,17]

(16 * 10 + 2) * 4 = 648

5) Adicione regras ao conjunto de regras da Fig. 6.36 (apresentada a seguir) para os seguintes comandos de controle de fluxo:

a) Comando $repeat$: $repeat\ S\ while\ B$

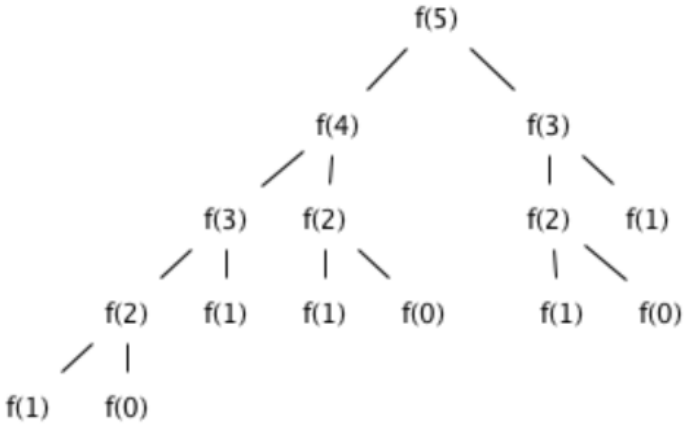
```
S1.next = newlabel()
B.true = newlabel()
B.false = S.next
S.code = label(B.true) || S1.code || label(S1.next) || B.code
```

b) Comando for: $for\ (S_1;\ B\ S_2)\ S_3$

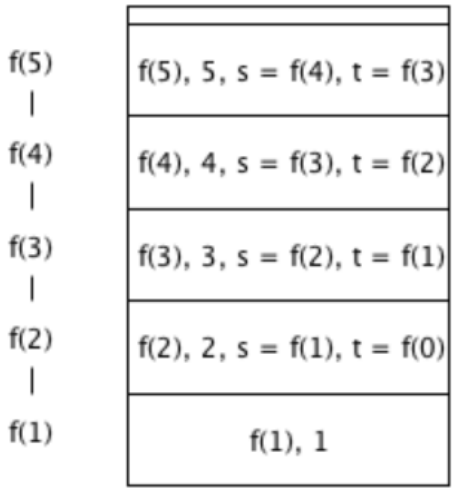
```
S1.next = newlabel()
B.true = newlabel()
B.false = S.next
S2.next = S1.next
S3.next = newlabel()
S.code = S1.code|| lable(S1.next) || B.code || lable(B.true) || S3.code ||
label(S3.next) || S2.code || gen('goto', S1.next)
```

6) O programa escrito em C abaixo computa os n´umero de Fibonacci recursivamente. Suponha que o registro de ativação para f inclui os seguintes elementos na seguinte ordem: (valor de retorno, argumento n, variável local s, variável local t); também há normalmente outros elementos no registro de ativação. As questões abaixo assumem que a chamada inicial é $f(5)$:

a) Mostre a árvore de ativação completa:



b) Como ficam a pilha de execução e os registros de ativação quando $f(1)$ está para retornar pela primeira vez?



c) Como ficam a pilha de execução e os registros de ativação quando $f(1)$ está para retornar pela quinta vez?