编译原理作业 12

23.11.22

6.4.3 (2)

假定 a 每行的宽度为 `wa1`, 元素宽度为 `wa2`; b 每行的宽度为 `wb1`, 元素宽度为 `wb2`.

```
t1 = i * wa1
t2 = j * wa2
t3 = t1 + t2
t4 = a[t3]
t5 = i * wb1
t6 = j * wb2
t7 = t5 + t6
t8 = b[t7]
t9 = t4 + t8
x = t9
```

6.4.6 (2)

(10-1)(20*4)+(8-1)*4=748.

6.4.8 (2)

0+(2-0)(6*8)+(7-5)*8=112.

6.5.2

产生式	语义规则
S -> id = E	E.unique = id.type ∈ E.type ? id.type : error()
E -> E_1 + E_2	E.type = E_1.type ∩ E_2.type E_1.unique = E.unique E_2.unique = E.unique
E -> - E_1	E.type = E_1.type E_1.unique = E.unique
E -> (E_1)	E.type = E_1.type E_1.unique = E.unique
E -> id	E.type = { id.type }
E -> E_1 (E_2)	E.type = { t | s ∈ E_2.type ∧ s→t ∈ E_1.type } E_2.unique = { s ∈ E_2.type | s→E.unique ∈ E_1.type } is singleton ? s : error() E_1.unique = E_2.unique→E.unique

6.6.1

产生式	语义规则
S -> repeat S_1 while B	begin = newlabel() S_1.next = newlabel() B.true = begin B.false = S.next S.code = gen('label' begin) || S_1.code || gen('label' S_1.next) || B.code
S -> for (S_1 ; B ; S_2) S_3	begin = newlabel() end = newlabel() S_1.next = end S_2.next = end S_3.next = newlabel() B.true = begin B.false = S.next S.code = S_1.code || gen('label' end) || B.code || gen('label 'begin') || S_3.code || gen('label' S_3.next) || S_2.code || gen('goto' end)

(1) 的结构示意图:

```
begin/B.true:
    S1
S1.next:
    B
S.next/B.false:
```

(2) 的结构示意图:

```
    S1
end/S1.next/S2.next:
    B
begin/B.true:
    S3
S3.next:
    S2
    goto end
S.next/B.false:
```

6.7.3

S_4.nextlist = E_3.falselist

S_5.nextlist = merge(E_4.falselist, S_2.nextlist)

S_6.nextlist = S_3.nextlist

S_7.nextlist = merge(S_4.nextlist, S_6.nextlist)

S_8.nextlist = E_1.falselist