5.21.

(a)   1*2*3 : { int , complex }

```
            /        |              \
          /          |                \
1*2:{int,complex}  *:{ int * int→ int,          3{int, complex}
     /|\              int * int → complex,
    / | \             complex*complex→complex}
```

            2:{int, complex}
int. *:{ int*int→ int,
omplex}int * int→ complex,

    complex * complex→ complex}   不唯一.

    结条可能为 int 或 complex.

(b)   1*(2*2) : { complex}

```
        /        |              \
      /          |                \
1:[complex]   *:{ int * int → int,      2*2 : {complex}
                  int * int → complex          /|\
                                              / | \
                  complex * complex →    2:[complex]*:  2:{int,complex}
                                                        {int*int→int,
                  complex}                               int*int→complex,
                                                         complex*complex
                                                         → complex}
```

    唯一, 结条总分是 complex.

(c)   (1 * z) * t : {complex}

(1*z):{complex};*:{int * int → int,
              int * int → complex,
              complex * complex → complex}
{int,
complex}   z:{complex}
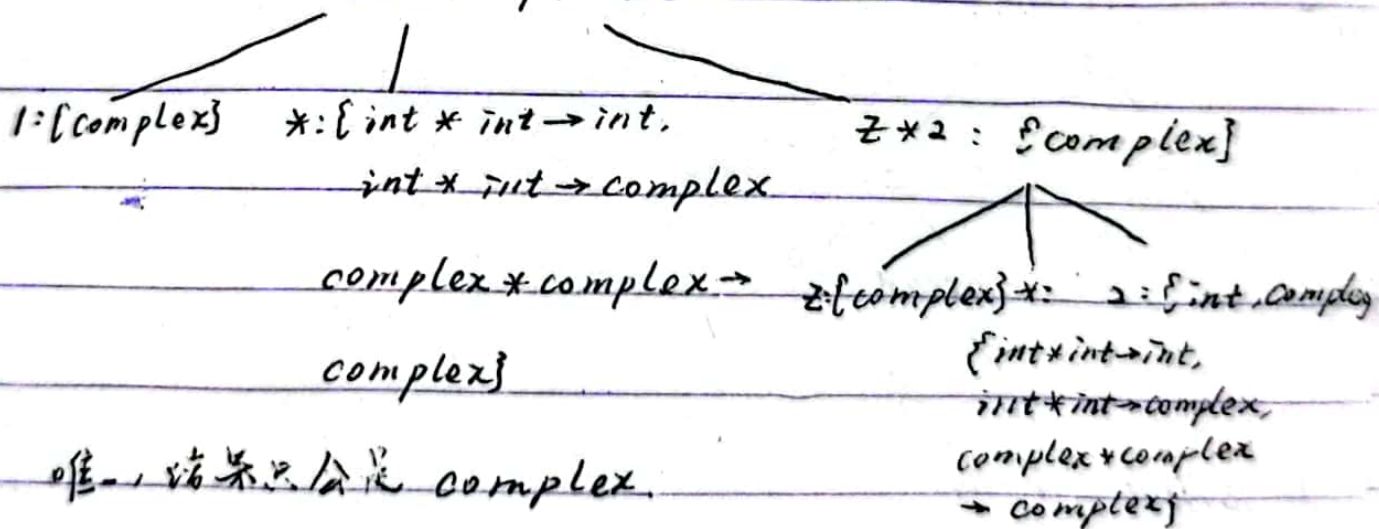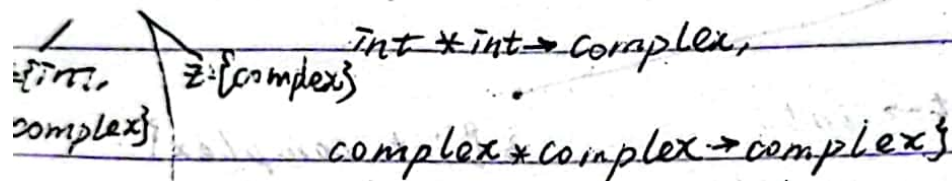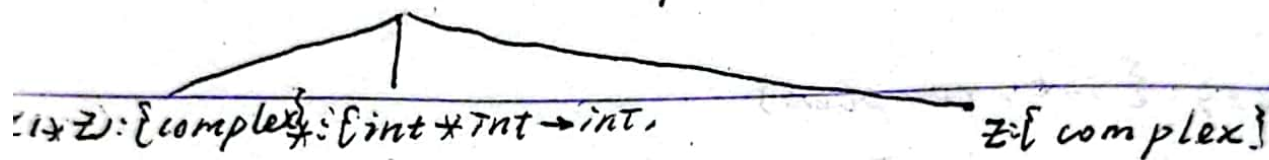
*:{int * int → int
   int * int → complex
   complex * complex → complex} .

唯一, 只可能为 complex

(a) 语法树

7.2.   while
                /   \
              ≤   =
             / \   / \
            i 10 array 0
                    / \
                   a   i

(b) 后缀表达式为

    i 10 ≤ 数组 a i array 0 = while

z:{ complex}
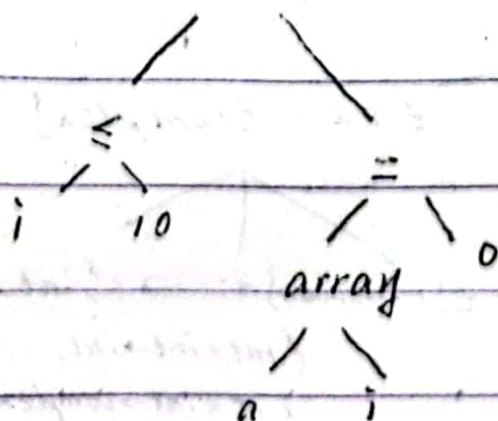
(c) Judge: if   i ≤ 10 goto Begin

        goto End

Begin: a[i] = 0

        goto Judge

End: return 0