输入输出说明：

执行本程序只需要运行 main.py 文件即可，运行本程序，将会看见两个**语法树**和其对应**结果**（分别对应**二分查找**和**选择排序**）。本次程序解析的两个文件分别为：binary_search.py 和 select_sort.py，已经分别在 main.py 文件中默认输入，

```
text=clear_text(open('binary_search.py','r').read())
text1=clear_text(open('select_sort.py', 'r').read())
```

如需改变输入，则改变对应解析文件中的内容即可。

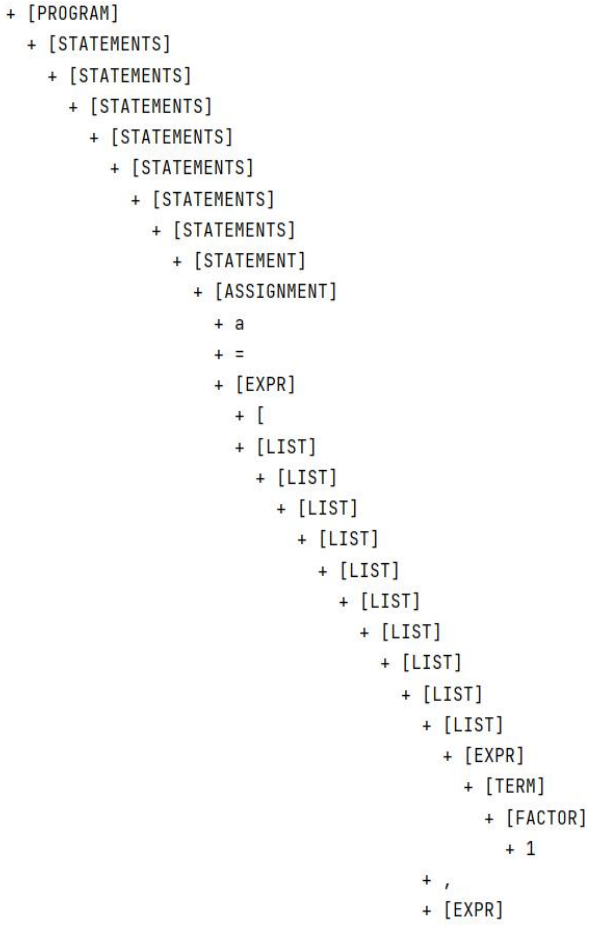
解析样例如下：

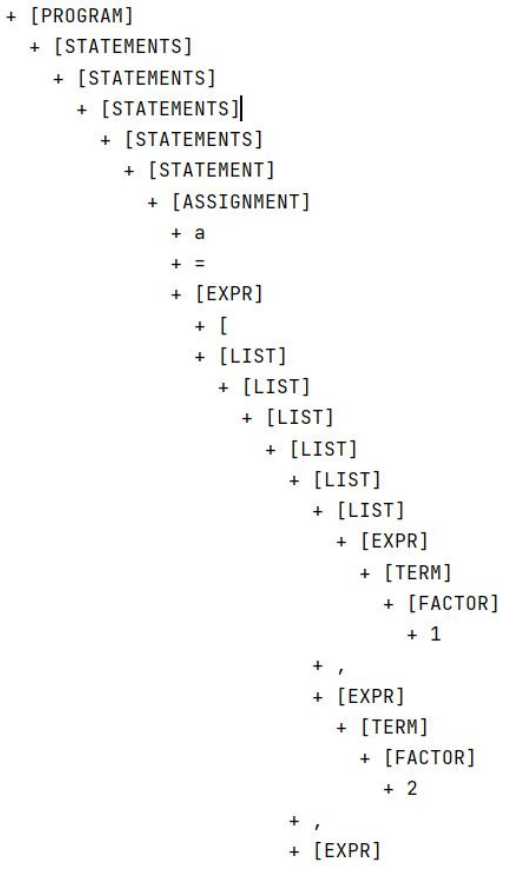
```
a=[1,2,3,4,5,6,7,8,9,10]
key=3

n=len(a)

begin=0
end=n-1

while(begin<=end){
    mid=(begin+end)//2
    if(a[mid]>key){
        end=mid-1
    }
    elif(a[mid]<key){
        begin=mid+1
    }
    else{
        break
    }
}
print(mid)
```

```
a=[1,2,4,3,6,5]

n=len(a)

for(i=0;i<n;i++){
    max_v=a[i]
    i_v=i

    for(j=i;j<n;j++){
        if(a[j]>max_v){
            max_v=a[j]
            i_v=j
        }
    }

    t=a[i]
    a[i]=a[i_v]
    a[i_v]=t
}

print(a)
```

程序输出结果如下：

```
2
{'a': [1, 2, 3, 4, 5, 6, 7, 8, 9, 10], 'key': 3, 'n': 10, 'begin': 2, 'end': 3, 'mid': 2}

[6, 5, 4, 3, 2, 1]
{'a': [6, 5, 4, 3, 2, 1], 'key': 3, 'n': 6, 'begin': 2, 'end': 3, 'mid': 2, 'i': 6, 'max_v': 1, 'i_v': 5, 'j': 6, 't': 1}
```


+ [PROGRAM]
+ [STATEMENTS]
+ [STATEMENTS]
+ [STATEMENTS]
+ [STATEMENTS]
+ [STATEMENTS]
+ [STATEMENTS]
+ [STATEMENTS]
+ [STATEMENT]
+ [ASSIGNMENT]
+ a
+ =
+ [EXPR]
+ [
+ [LIST]
+ [LIST]
+ [LIST]
+ [LIST]
+ [LIST]
+ [LIST]
+ [LIST]
+ [LIST]
+ [LIST]
+ [EXPR]
+ [TERM]
+ [FACTOR]
+ 1
+ ,
+ [EXPR]


+ [PROGRAM]
+ [STATEMENTS]
+ [STATEMENTS]
+ [STATEMENTS]
+ [STATEMENTS]
+ [STATEMENT]
+ [ASSIGNMENT]
+ a
+ =
+ [EXPR]
+ [
+ [LIST]
+ [LIST]
+ [LIST]
+ [LIST]
+ [LIST]
+ [LIST]
+ [EXPR]
+ [TERM]
+ [FACTOR]
+ 1
+ ,
+ [EXPR]
+ [TERM]
+ [FACTOR]
+ 2
+ ,
+ [EXPR]