编译原理第一次实验测试用例:目录

1	A 组测试用例	2
	1.1 A-1	2
	1.2 A-2	2
	1.3 A-3	3
	1.4 A-4	3
	1.5 A-5	4
	1.6 A-6	4
	1.7 A-7	5
	1.8 A-8	5
	1.9 A-9	6
2	B组测试用例	7
	2.1 B-1	7
	2.2 B-2	8
3	C 组测试用例	9
	3.1 C-1	10
	3.2 C-2	14
4	D组测试用例	22
	4.1 D-1	22
	4.2 D-2	25
	4.3 D-3	29
5	E 组测试用例	33
	5.1 E1.1	33
	5.2 E1.2	34
	5.3 E1.3	35
6	结束语	37

1 A 组测试用例

本组测试用例共9个,每个仅包含单个的词法或者语法错误。除特殊说明外,不可多报。多报、漏报错误,或者打印语法树都会导致扣分。错误编号和行号之后的说明文字不要求与给出的输出完全一致,仅供助教理解使用,不作为评分依据。

1.1 A-1

输入

```
float _func1(int X1)

float a_1_2;

int 6_Wrong;

return 0;

}
```

输出

```
Error type B at line 4: syntax error, near ';'
```

说明:错误类型也可以是 A 类,或者一个 A 一个 B,但是只能在第 4 行。这里有一个非法标识符 6_Wrong,注意标识符可以以下划线开始,所以第 1 行正确。

1.2 A-2

输入

```
int sub(int a, int b)

int c = a + b - a * b / 3;

int d = !a && b || c >= 4;

int e = a & b;

return e;

}
```

```
Error type A at line 5: mysteriously character '&'
```

说明:必须有 type A 错误;可以多报一个 type B 错误。这里有一个非法的符号 &。

1.3 A-3

输入

```
int add()

int a, b, c;

c = (a + b) - (c * d / (-a)));

b = a || c;

return b;

}
```

输出

```
Error type B at line 4: syntax error
```

说明:第4行缺少匹配的左括号。

1.4 A-4

输入

```
int;
int;
int if()

int a;
if (a > 0)
    return 1;
else
    return 0;

}
```

输出

```
Error type B at line 2: syntax error
```

说明:第2行保留字if不能作为标识符使用。注意第1行语句没有意义但是合法。

1.5 A-5

输入

```
int array(int i)

int array(int i)

int a[3][3*3];

a[i][i] = 1;

while (i < 9)

{
    a[0][i] = 1;
    i = i + 1;
}

}</pre>
```

输出

```
Error type B at line 3: syntax error, near '*'
```

说明: 第3行数组定义格式错误, 出现表达式。

1.6 A-6

输入

```
int a = 1;
int add(int b)

{
    return a + b;
}
int main()

{
    int sum;
    sum = add(2);
}
```

Error type B at line 1: syntax error, the global variable cannot be initialized in the definition.

说明: 第1行定义全局变量时进行了初始化。

1.7 A-7

输入

```
struct Date
{
    int year, month;
    int day;
};
int main()
{
    Date d[10];
    d[0].year = 2020;
}
```

输出

```
Error type B at line 8: syntax error, missing 'struct'
```

说明:第8行定义结构体时缺少关键字 struct。

1.8 A-8

输入

```
int main()

{
    struct A

4    {
        int a;
        float b;
    };
```

```
struct A a[100];
int i = 0;
while (i < 100)
{
    a[i].a = a[i].b = 0;
    i = i + 1;
}
</pre>
```

```
Error type B at line 7: syntax error, near ';'
```

说明: 第7行, 在方法体内定义结构体时缺少声明的变量名, 报在第3行也可以。

1.9 A-9

输入

```
int max(int x, int y)
2
       int t;
3
       if (x > y)
           t = x;
       else
          t = y
       return t;
8
9
  int main()
11
       int a = 6, b = 4, c = 5;
12
       int maxs = max(a, max(b, c));
13
14
```

```
Error type B at line 7: syntax error, missing ';'
```

说明: 第7行漏写了句尾的分号。也可在第8行报错。

2 B组测试用例

本组测试用例共2个,每个用例包含多处不同的错误。除特殊说明外,漏报、多报错误或者打印语法树都会导致扣分。

2.1 B-1

输入

```
struct Student
2
      int ID,
3
      float score;
  }class[2];
  struct Student test(struct Student a)
8
      return a;
10
  int main()
12
13
       int i = 0, n = 2;
14
       float sum = 0;
       struct Student classes[n][10];
16
      class[0].ID = 1;
17
       class[0].score = 90;
18
       class[1].ID = 2;
       class[1].score = .85 * 100;
       test(class[0]);
21
      while (i < 2)
22
23
           sum += class[i].score;
24
```

```
Error type B at line 3: syntax error, near ','

Error type B at line 16: syntax error, near '['

Error type A at line 20: illegal float number '.85'

Error type B at line 24: syntax error, near '+'
```

说明:第3行末尾分号错写成逗号,也可以报错在第4行;第16行用变量定义数组;第20行小数点前必须有数字,也可识别成B类型错误,或多报一个B类错误;第24行使用了未定义的操作符+=。

2.2 B-2

输入

```
int shsort(int s[4], int n)
       int i, j, d;
3
       d = n / 2;
4
       while (d > = 1)
            i = d + 1;
           while (i <= n)</pre>
            {
                s[0] = s[i];
10
                j = i - d;
11
                while((j > 0) \&\& (s[0] < s[j]))
12
13
                     s[j + d] = s[j];
                     j = j - d;
15
                }
16
```

```
s[j + d] = s[0];
17
                  i ++;
19
             }
             d = d / 2;
20
        }
21
        return 0;
22
23
   int main()
24
25
        int a[4];
26
        a[0] = 2;
        a[1] = 4;
28
        a[2] = 3;
29
        a[3]] = 1;
30
        int n = 4;
31
        shsort(a, n);
32
        return 0;
33
34
```

```
Error type B at line 5: syntax error, near '>'
Error type B at line 18: syntax error, near '+'
Error type B at line 30: syntax error, near ']'
Error type B at line 31: syntax error, the local variable cannot be defined after statements
```

说明: 第5行>=号中间多了一个空格; 第18行使用了未定义符号++; 第30行多打了一个右括号]; 第31行语句之后不能再出现变量定义。

3 C组测试用例

本组测试用例共2个,不包含任何错误,需要输出正确的语法树。除特殊说明外,应与给出的语法树完全相同。语法树打印错误酌情扣分。

3.1 C-1

输入

```
struct Student
2
       int SID;
3
       int age;
       struct Teacher
6
          int TID;
       } t;
  };
10
  int main()
11
12
       struct Student s1;
13
       s1.SID = 1;
14
       s1.age = 18;
15
       s1.t.TID = 3;
       return s1;
17
18
```

```
Program (1)
ExtDefList (1)

ExtDef (1)

Specifier (1)

StructSpecifier (1)

STRUCT

OptTag (1)

ID: Student

LC

DefList (3)
```

```
Def (3)
                   Specifier (3)
12
                     TYPE: int
13
                   DecList (3)
14
                     Dec (3)
15
                       VarDec (3)
                          ID: SID
17
                   SEMI
18
                 DefList (4)
19
                   Def (4)
20
                     Specifier (4)
21
                        TYPE: int
22
                     DecList (4)
23
                        Dec (4)
24
                          VarDec (4)
25
                            ID: age
26
                     SEMI
27
                   DefList (5)
28
                     Def (5)
                        Specifier (5)
30
                          StructSpecifier (5)
31
                             STRUCT
32
                             OptTag (5)
33
                               ID: Teacher
34
                             LC
35
                             DefList (7)
36
                               Def (7)
                                 Specifier (7)
38
                                   TYPE: int
39
                                 DecList (7)
40
                                    Dec (7)
41
                                      VarDec (7)
```

```
ID: TID
43
                                 SEMI
44
                            RC
45
                        DecList (8)
46
                          Dec (8)
47
                            VarDec (8)
                              ID: t
                        SEMI
50
              RC
51
          SEMI
52
       ExtDefList (11)
53
         ExtDef (11)
54
            Specifier (11)
55
              TYPE: int
56
            FunDec (11)
57
              ID: main
58
              LΡ
59
              RP
60
            CompSt (12)
61
              LC
62
              DefList (13)
63
                 Def (13)
64
                   Specifier (13)
                     StructSpecifier (13)
66
                       STRUCT
67
                       Tag (13)
68
                         ID: Student
                   DecList (13)
                     Dec (13)
71
                       VarDec (13)
72
                          ID: s1
73
                   SEMI
```

```
StmtList (14)
75
                 Stmt (14)
                   Exp (14)
77
                     Exp (14)
78
                       Exp (14)
79
                        ID: s1
                       DOT
81
                       ID: SID
82
                     ASSIGNOP
83
                     Exp (14)
84
                      INT: 1
                   SEMI
86
                 StmtList (15)
87
                   Stmt (15)
88
                     Exp (15)
89
                       Exp (15)
90
                          Exp (15)
91
                           ID: s1
92
                         DOT
                         ID: age
                       ASSIGNOP
95
                       Exp (15)
96
                         INT: 18
97
                     SEMI
98
                   StmtList (16)
99
                     Stmt (16)
100
                       Exp (16)
                          Exp (16)
102
                            Exp (16)
103
                              Exp (16)
104
                                ID: s1
```

DOT

106

```
ID: t
107
                                DOT
108
                                ID: TID
109
                             ASSIGNOP
110
                             Exp (16)
111
                                INT: 3
112
                           SEMI
113
                        StmtList (17)
114
                           Stmt (17)
115
                             RETURN
                             Exp (17)
117
                                ID: s1
118
                              SEMI
119
                RC
120
```

说明:使用的空格可以用 Tab 替换,注意缩进

3.2 C-2

输入

```
int MatrixMax(int a[3][4])
2
       int i=0,j=0;
3
       int max, max_i=0, max_j=0;
4
       \max = a[0][0];
       while(i < 3)
6
       {
           while(j < 4)
                if(a[i][j] > max)
10
11
                    max=a[i][j];
12
                    max_i=i;
13
                    max_j=j;
14
```

```
Program (1)
     ExtDefList (1)
2
       ExtDef (1)
         Specifier (1)
            TYPE: int
5
         FunDec (1)
           ID: MatrixMax
           LP
           VarList (1)
              ParamDec (1)
10
                Specifier (1)
11
                  TYPE: int
                VarDec (1)
13
                  VarDec (1)
14
                    VarDec (1)
15
                     ID: a
                     LB
17
                     INT: 3
18
                    RB
19
                  LB
                  INT: 4
21
                  RB
22
           RP
23
         CompSt (2)
```

```
LC
25
            DefList (3)
26
              Def (3)
27
                 Specifier (3)
28
                   TYPE: int
29
                 DecList (3)
                   Dec (3)
31
                     VarDec (3)
32
                       ID: i
33
                     ASSIGNOP
34
                     Exp (3)
35
                       INT: 0
36
                   COMMA
37
                   DecList (3)
38
                     Dec (3)
39
                       VarDec (3)
40
                          ID: j
41
                       ASSIGNOP
42
                       Exp (3)
43
                         INT: 0
44
                 SEMI
45
              DefList (4)
46
                 Def (4)
47
                   Specifier (4)
48
                     TYPE: int
49
                   DecList (4)
50
                     Dec (4)
                       VarDec (4)
52
                          ID: max
53
                     COMMA
54
                     DecList (4)
55
                       Dec (4)
```

```
VarDec (4)
57
                           ID: max_i
                         ASSIGNOP
59
                         Exp (4)
60
                          INT: 0
61
                      COMMA
                       DecList (4)
63
                         Dec (4)
64
                          VarDec (4)
65
                            ID: max_j
                           ASSIGNOP
                           Exp (4)
68
                             INT: 0
69
                  SEMI
70
           StmtList (5)
71
             Stmt (5)
72
                Exp (5)
73
                  Exp (5)
74
                   ID: max
75
                  ASSIGNOP
76
                  Exp (5)
77
                    Exp (5)
78
                      Exp (5)
                        ID: a
80
                      LB
81
                     Exp (5)
82
                      INT: 0
                     RB
84
                    LB
85
                    Exp (5)
86
                     INT: 0
87
                    RB
```

```
SEMI
89
                StmtList (6)
90
                  Stmt (6)
91
                    WHILE
92
                    LΡ
93
                    Exp (6)
94
                      Exp (6)
95
                        ID: i
96
                       RELOP
97
                      Exp (6)
                        INT: 3
                    RP
100
                     Stmt (7)
101
                       CompSt (7)
102
                         LC
103
                         StmtList (8)
104
                            Stmt (8)
105
                              WHILE
106
                              LP
107
                              Exp (8)
108
                                Exp (8)
109
                                  ID: j
110
                                 RELOP
111
                                 Exp (8)
112
                                   INT: 4
113
                               RP
114
                               Stmt (9)
115
                                 CompSt (9)
116
                                   LC
117
                                   StmtList (10)
118
                                      Stmt (10)
119
                                        ΙF
120
```

121	LP
122	Exp (10)
123	Exp (10)
124	Exp (10)
125	Exp (10)
126	ID: a
127	LB
128	Exp (10)
129	ID: i
130	RB
131	LB
132	Exp (10)
133	ID: j
134	RB
135	RELOP
136	Exp (10)
137	ID: max
138	RP
139	Stmt (11)
140	CompSt (11)
141	LC
142	StmtList (12)
143	Stmt (12)
144	Exp (12)
145	Exp (12)
146	ID: max
147	ASSIGNOP
148	Exp (12)
149	Exp (12)
150	Exp (12)
151	ID: a
152	LB

153	Exp (12)
154	ID: i
155	RB
156	LB
157	Exp (12)
158	ID: j
159	RB
160	SEMI
161	StmtList (13)
162	Stmt (13)
163	Exp (13)
164	Exp (13)
165	ID: max_i
166	ASSIGNOP
167	Exp (13)
168	ID: i
169	SEMI
170	StmtList (14)
171	Stmt (14)
172	Exp (14)
173	Exp (14)
174	ID: max_j
175	ASSIGNOP
176	Exp (14)
177	ID: j
178	SEMI
179	RC
180	StmtList (16)
181	Stmt (16)
182	Exp (16)
183	Exp (16)
184	ID: j

```
ASSIGNOP
185
                                              Exp (16)
186
                                                 Exp (16)
187
                                                  ID: j
188
                                                 PLUS
189
                                                 Exp (16)
190
                                                   INT: 1
191
                                            SEMI
192
                                    RC
193
                             StmtList (18)
194
                               Stmt (18)
                                  Exp (18)
196
                                    Exp (18)
197
                                      ID: i
198
                                    ASSIGNOP
                                    Exp (18)
200
                                      Exp (18)
201
                                        ID: i
202
                                      PLUS
203
                                      Exp (18)
204
                                         INT: 1
205
                                  SEMI
206
                          RC
207
                  StmtList (20)
208
                     Stmt (20)
209
                       RETURN
210
                       Exp (20)
                         INT: 0
212
                       SEMI
213
             RC
214
```

说明:考察对数组的翻译。

4 D组测试用例

本组测试用例共 3 个,针对不同分组进行测试。对应分组的同学需要输出语法树,提示错误则不得分;其他分组的同学只需要在对应位置提示错误即可,如果打印了语法树,则将视为违规,将会倒扣分。

4.1 D-1

输入

```
int func_test()

int _dec_ = 947;

int _oct_ = 0705;

int _dhex_ = 0xFFaBc - _oct_;

int _result_ = - _dhex_ + _oct_ * ( _dec_ - 0x23fD );

}
```

```
Program (1)
    ExtDefList (1)
2
       ExtDef (1)
3
         Specifier (1)
           TYPE: int
5
         FunDec (1)
           ID: func_test
           LΡ
8
           RP
         CompSt (2)
10
           LC
           DefList (3)
12
             Def (3)
13
                Specifier (3)
14
                  TYPE: int
                DecList (3)
16
```

```
Dec (3)
17
                     VarDec (3)
                      ID: _dec_
19
                     ASSIGNOP
20
                     Exp (3)
21
                      INT: 947
22
                SEMI
23
              DefList (4)
24
                Def (4)
25
                   Specifier (4)
26
                     TYPE: int
                   DecList (4)
28
                     Dec (4)
29
                       VarDec (4)
30
                         ID: _oct_
31
                       ASSIGNOP
32
                      Exp (4)
33
                         INT: 453
34
                   SEMI
35
                DefList (5)
36
                   Def (5)
37
                     Specifier (5)
38
                       TYPE: int
39
                     DecList (5)
40
                       Dec (5)
41
                         VarDec (5)
42
                            ID: _dhex_
43
                         ASSIGNOP
44
                          Exp (5)
45
                            Exp (5)
46
                             INT: 1047228
47
                            MINUS
```

```
Exp (5)
49
                            ID: _oct_
50
                     SEMI
51
                   DefList (6)
52
                     Def (6)
53
                       Specifier (6)
                         TYPE: int
55
                       DecList (6)
56
                         Dec (6)
57
                           VarDec (6)
58
                             ID: _result_
                           ASSIGNOP
60
                            Exp (6)
61
                             Exp (6)
62
                                MINUS
63
                               Exp (6)
64
                               ID: _dhex_
65
                              PLUS
66
                              Exp (6)
                                Exp (6)
68
                                 ID: _oct_
69
                                STAR
70
                                Exp (6)
71
                                  LP
72
                                  Exp (6)
73
                                    Exp (6)
74
                                     ID: _dec_
                                    MINUS
76
                                    Exp (6)
77
                                      INT: 9213
78
                                  RP
79
                       SEMI
```

81 RC

说明: 1.1 分组的同学需要输出该语法树, 8 进制和 16 进制数必须正确转换(453、1047228和 9213); 其他分组的同学只要提示相应的错误,而且不输出语法树即可。

4.2 D-2

输入

```
int float_test()

{
    float X_1 = 3.80e-7;
    float X_2 = 345.2e+4;

    float X_3 = 2.76E+3;

    float X_4 = .994E-2;

    float X_5 = 11.e1;

    float X_6 = -0.4E-03;

    float result = (15.E-1 + X_3) + X_6;

}
```

```
Program (1)
    ExtDefList (1)
2
      ExtDef (1)
         Specifier (1)
           TYPE: int
5
         FunDec (1)
           ID: float test
           LΡ
8
           RP
         CompSt (2)
10
           LC
11
           DefList (3)
             Def (3)
13
                Specifier (3)
14
```

```
TYPE: float
                DecList (3)
                  Dec (3)
17
                    VarDec (3)
18
                      ID: X_1
19
                     ASSIGNOP
                    Exp (3)
21
                      FLOAT: 0.000000
22
                SEMI
23
              DefList (4)
24
                Def (4)
                  Specifier (4)
26
                     TYPE: float
27
                  DecList (4)
28
                     Dec (4)
                       VarDec (4)
30
                         ID: X_2
31
                       ASSIGNOP
32
                       Exp (4)
33
                        FLOAT: 3452000.000000
34
                   SEMI
35
                DefList (5)
36
                  Def (5)
37
                     Specifier (5)
38
                       TYPE: float
39
                     DecList (5)
40
                       Dec (5)
41
                         VarDec (5)
42
                           ID: X 3
43
                         ASSIGNOP
44
                         Exp (5)
45
                           FLOAT: 2760.000000
```

```
SEMI
47
                   DefList (6)
48
                     Def (6)
                       Specifier (6)
50
                         TYPE: float
51
                       DecList (6)
                         Dec (6)
53
                           VarDec (6)
54
                             ID: X_4
55
                           ASSIGNOP
                           Exp (6)
                             FLOAT: 0.009940
58
                       SEMI
59
                     DefList (7)
60
                       Def (7)
61
                         Specifier (7)
62
                           TYPE: float
63
                         DecList (7)
64
                           Dec (7)
                             VarDec (7)
66
                                ID: X 5
67
                              ASSIGNOP
68
                              Exp (7)
                               FLOAT: 110.000000
70
                         SEMI
71
                       DefList (8)
72
                         Def (8)
                            Specifier (8)
                              TYPE: float
75
                           DecList (8)
76
                              Dec (8)
77
                               VarDec (8)
```

```
ID: X 6
79
                                   ASSIGNOP
80
                                   Exp (8)
81
                                     MINUS
82
                                     Exp (8)
83
                                       FLOAT: 0.000400
                              SEMI
85
                            DefList (9)
86
                              Def (9)
87
                                 Specifier (9)
                                   TYPE: float
                                 DecList (9)
90
                                   Dec (9)
91
                                     VarDec (9)
92
                                        ID: result
93
                                      ASSIGNOP
94
                                      Exp (9)
95
                                        Exp (9)
96
                                          LΡ
                                          Exp (9)
98
                                             Exp (9)
99
                                               FLOAT: 1.500000
100
                                             PLUS
101
                                             Exp (9)
102
                                               ID: X_3
103
                                           RP
104
                                        PLUS
105
                                        Exp (9)
106
                                           ID: X 6
107
                                 SEMI
108
             RC
109
```

说明: 1.2 分组的同学需要输出语法树,注意科学计数法浮点数的正确转换。其它分组同学

只需要提示相应错误,而且不输出语法树即可。

4.3 D-3

输入

```
**This is a test for comments
2
  int /*/**/F(int n) // {%^@#~~~ //
7
      /***8
8
       int i = 0;
       ****/*/
10
       if(n == 0)
11
12
           return 1; /*} \\\
13
14
15
      return n * F(n - 1);
17
  int main() //b > a comments*//\//*
18
19
       int n, r;
20
      n = 10;
21
      r = F(n); // \ *// \ *// \ *// \ *// \ 
22
       return /*ends\/\/\/\//*/0;
23
```

```
Program (6)
ExtDefList (6)
ExtDef (6)
```

```
Specifier (6)
            TYPE: int
          FunDec (6)
6
            ID: F
            LΡ
8
            VarList (6)
              ParamDec (6)
10
                 Specifier (6)
11
                   TYPE: int
12
                 VarDec (6)
13
                   ID: n
14
            RP
15
          CompSt (7)
16
            LC
17
            StmtList (11)
18
               Stmt (11)
19
                 ΙF
20
                 LΡ
21
                 Exp (11)
22
                   Exp (11)
23
                      ID: n
24
                   RELOP
25
                   Exp (11)
                      INT: 0
27
                 RP
28
                 Stmt (13)
29
                   RETURN
                   Exp (13)
31
                      INT: 1
32
                   SEMI
33
               StmtList (16)
34
                 Stmt (16)
```

```
RETURN
36
                   Exp (16)
37
                      Exp (16)
38
                       ID: n
39
                      STAR
40
                      Exp (16)
41
                        ID: F
42
                        LP
43
                        Args (16)
44
                           Exp (16)
45
                             Exp (16)
46
                               ID: n
47
                             MINUS
48
                             Exp (16)
49
                               INT: 1
50
                       RP
51
                   SEMI
52
            RC
53
       ExtDefList (18)
          ExtDef (18)
55
            Specifier (18)
56
              TYPE: int
57
            FunDec (18)
58
               ID: main
59
              LΡ
60
               RP
61
            CompSt (19)
               LC
63
               DefList (20)
64
                 Def (20)
65
                   Specifier (20)
                      TYPE: int
```

```
DecList (20)
68
                    Dec (20)
                      VarDec (20)
70
                        ID: n
71
                    COMMA
72
                    DecList (20)
                      Dec (20)
                        VarDec (20)
75
                          ID: r
76
                  SEMI
77
              StmtList (21)
78
                Stmt (21)
                  Exp (21)
80
                    Exp (21)
81
                     ID: n
82
                    ASSIGNOP
83
                   Exp (21)
84
                     INT: 10
85
                  SEMI
                StmtList (22)
87
                  Stmt (22)
88
                    Exp (22)
89
                      Exp (22)
                         ID: r
91
                      ASSIGNOP
92
                      Exp (22)
93
                         ID: F
                         LP
                         Args (22)
96
                          Exp (22)
97
                            ID: n
                         RP
```

```
SEMI
StmtList (23)
Stmt (23)
RETURN
Exp (23)
INT: 0
SEMI
RC
```

说明: 1.3 分组的同学需要输出语法树,不能提示有语法错误;其他分组同学只需要提示相应错误,且不输出语法树即可。

5 E 组测试用例

本组测试用例共6个,针对不同分组进行测试。

5.1 E1.1

这组测试用例针对1.1分组的同学。

输入(E1-1)

```
int test_for_wrong_oct_number()

int _correct_oct_number_ = 003456;

int _decimal_number = 748;

int _wrong_oct_number_ = 0748;

return 0;

}
```

输出

```
Error type A at Line 5: Illegal octal number "0748"
```

说明: 仅 1.1 分组的同学需要测试这个用例,针对错误的 8 进制数 0748,识别成错误类型 B 也可以。

输入(E1-2)

```
int test_for_wrong_dhex_number()

int _correct_dhex_number_ = 0xf29C0;

int _wrong_dhex_number_ = 0xCd0G;

int _correct_dhex_number2_ = 0x000abCD;

int _wrong_dhex_number2_ = 0xx346f;

}
```

```
Error type A at Line 4: Illegal hexadecimal number "0xCd0G"

Error type A at Line 6: Illegal hexadecimal number "0xx346f"
```

说明: 仅 1.1 分组的同学需要测试这个用例,针对错误的 16 进制数 0xCd0G 与 0xx346f,识别成错误类型 B 也可以。

5.2 E1.2

这组测试用例针对 1.2 分组的同学。

输入(E2-1)

```
float function()

float x1 = 002.45E10;

float x2 = 45.e-1.1;

float x3 = .45e-3;

float x4 = 34.34E+1.2;

return x1 + x3;

}
```

输出

```
Error type A at Line 4: Illegal float number "45.e-1.1"

Error type A at Line 6: Illegal float number "34.34E+1.2"
```

说明: 仅 1.2 分组的同学需要测试这个用例,针对错误浮点数 45.e-1.1 和 34.34E+1.2,识别成错误类型 B 也可以。

输入(E2-2)

```
float function()

float a = 12.34e+;

float b = .e;

float c = .e-2;

}
```

输出

```
Error type A at Line 3: Illegal float number "12.34e+"

Error type B at line 4: syntax error, unexpected DOT

Error type A at Line 5: Illegal float number ".e-2"
```

说明:仅 1.2 分组的同学需要测试这个用例,针对错误浮点数 12.34e+、.e 和.e-2,识别成错误类型 B 和 A 都可以。

5.3 E1.3

这组测试用例针对 1.3 分组的同学。

输入(E3-1)

```
* @param int a
  */
3
  int main()
5
      /* TODO
                  definition 12345
          */
8
      int day, x1, x2;
9
      day = 9 /* this is okay*/;
10
      x2 = 1;
      // \\\\///\\\\~~/////\\\\*/
12
      while(day> /*>=<*/ 0)
13
```

```
x1=(x2+1)*2;
15
               /////\\\\\\////>
17
               /* \\\\\\"''' this is not okay!! */
18
          * /
19
          x2=x1;
          day = day - 1;
21
       }// assert(day == ?);
22
       return /*-1;*/0;
23
24
```

```
Error type B at Line 19: syntax error, near '*'
```

说明: 仅 1.3 分组的同学需要测试这个用例,针对嵌套的多行注释。19 行出现了多余的注释符号。识别成类型 A 也可以,需要合理的错误提示。

输入(E3-2)

```
/***comment
  still
  */
3
  /* ///\\asfdlkajhsldf\\\<>@#$
  * /
  int main() //a function
7
      float h, s; int i;
8
      h = s = 100 //;
9
      h = h / 2; // first
10
       i = 2;
11
      /* a new comment*/
12
      while (i <= 10) //\\//\*</pre>
13
           s = s + 2 * h;
15
          h = h / 2 /* / / / / / / / / / / **** / * /;
```

```
i = i + 1;

i = i + 1;

return 0;

/**end ~??
```

```
Error type B at Line 9: missing ';'

Error type B at Line 21: syntax error, no match comment "/*"
```

说明: 仅 1.3 分组的同学需要测试这个用例。第 9 行语句漏写了分号,也可以报在第 10 行; 第 21 行第缺少结束的注释,如果打印了语法树,或者程序异常终止、死循环无法退出等,则该用例不得分。不限定错误类型以及提示方式,但是出错位置必须限定在 21 行或者以后的位置;直接提示"未终止的注释"也可以。

6 结束语

如果对本测试用例有任何疑议,可以写邮件与<mark>李聪</mark>助教或<mark>陈紫琦</mark>助教联系,注意同时抄送 给许老师。