Đã bắt đầu vào lúc	Tuesday, 12 October 2021, 8:02 AM
Tình trạng	Đã hoàn thành
Hoàn thành vào lúc	Monday, 18 October 2021, 10:57 PM
Thời gian thực hiện	6 ngày 14 giờ
Điểm	3,00/3,00
Điểm	**10,00** của 10,00 (**100**%)

Câu hỏi **1**

Chính xác

Điểm 1,00 của 1,00

Let AST of a programming language be defined as follows:

class Program: #decl:List[ClassDecl]

class Decl(ABC): #abstract class

class ClassDecl:#name:str,parent:str,mem:List[Decl]

class VarDecl(Decl): #name:str,typ:Type

class FuncDecl(Decl): #name:str,param:List[VarDecl],body:Tuple(List[Decl],List[Expr])

class Type(ABC): #abstract class

class IntType(Type)

class FloatType(Type)

class ClassType(Type):#name:str

class Expr(ABC): #abstract class

class Lit(Expr): #abstract class

class IntLit(Lit): #val:int

class Id(Expr): #name:str

and exceptions:

class RedeclaredField(Exception): #name:str

class UndeclaredMethod(Exception): #name:str

Implement the methods of the following class Visitor to travel on the above AST to detect undeclared declarations (throw the exception UndeclaredIdentifier). Note that the redeclared declarations exception also is thrown if a redeclared declaration is detected:

class StaticCheck(Visitor):

    def visitProgram(self,ctx:Program,o:object): pass

    def visitClassDecl(self,ctx:ClassDecl,o:object):pass

    def visitVarDecl(self,ctx:VarDecl,o:object):pass

    def visitFuncDecl(self,ctx:FuncDecl,o:object):pass

    def visitIntType(self,ctx:IntType,o:object):pass

    def visitFloatType(self,ctx:FloatType,o:object):pass

    def visitClassType(self,ctx:ClassType,o:object):pass

    def visitIntLit(self,ctx:IntLit,o:object):pass

    def visitId(self,ctx:Id,o:object):pass

    def visitFieldAccess(self,ctx:FieldAccess,o:object):pass

Your code starts at line 65

For example:

Test	Result
`Program([ClassDecl("x","",[VarDecl("a",IntType()),FuncDecl("a",[],([VarDecl("m",ClassType("x"))],[FieldAccess(Id("m"),"a"),FieldAccess(Id("m"),"b")]))])])`	`Redeclared Method: a`

Answer: (penalty regime: 0 %)

```
14
15 ▾    def visitVarDecl(self,ctx:VarDecl,o:object):
16 ▾        if ctx.name in o:
17              raise RedeclaredField(ctx.name)
18          o.append(ctx.name)
19
20 ▾    def visitFuncDecl(self,ctx:FuncDecl,o:object):
21          lst = []
22 ▾        if ctx.name in o:
23              raise RedeclaredMethod(ctx.name)
```

```
24          o.append(ctx.name)
25
26      def visitIntType(self,ctx:IntType,o:object):pass
27
28      def visitFloatType(self,ctx:FloatType,o:object):pass
29
30      def visitClassType(self,ctx:ClassType,o:object):pass
31
32      def visitIntLit(self,ctx:IntLit,o:object):pass
33
34      def visitId(self,ctx:Id,o:object):pass
35
36      def visitFieldAccess(self,ctx:FieldAccess,o:object):pass
```

	Test	Expected	Got	
✔	`Program([ClassDecl("x","",[VarDecl("a",IntType()),FuncDecl("a",[],([VarDecl("m",ClassType("x"))],[FieldAccess(Id("m"),"a"),FieldAccess(Id("m"),"b")]))])])`	`Redeclared Method: a`	`Redeclared Method: a`	✔
✔	`Program([ClassDecl("y","",[VarDecl("a",IntType())]),ClassDecl("x","y",[FuncDecl("a",[],([],[])),FuncDecl("b",[],([],[])),VarDecl("b",IntType())])])`	`Redeclared Field: b`	`Redeclared Field: b`	✔

Passed all tests! ✔

Chính xác

Điểm cho bài nộp này: 1,00/1,00.

Câu hỏi **2**

Chính xác

Điểm 1,00 của 1,00

Let AST of a programming language be defined as follows:

class Program: #decl:List[ClassDecl]

class Decl(ABC): #abstract class

class ClassDecl:#name:str,parent:str,mem:List[Decl]

class VarDecl(Decl): #name:str,typ:Type

class FuncDecl(Decl): #name:str,param:List[VarDecl],body:Tuple(List[Decl],List[Expr])

class Type(ABC): #abstract class

class IntType(Type)

class FloatType(Type)

class ClassType(Type):#name:str

class Expr(ABC): #abstract class

class Lit(Expr): #abstract class

class IntLit(Lit): #val:int

class Id(Expr): #name:str

class FieldAccess(Expr): #exp:Expr,field:str

and exceptions:

class UndeclaredIdentifier(Exception): #name:str

class UndeclaredField(Exception): #name:str

Implement the methods of the following class Visitor to travel on the above AST to detect undeclared declarations (throw the exception UndeclaredIdentifier). Note that the redeclared declarations exception also is thrown if a redeclared declaration is detected:

class StaticCheck(Visitor):

    def visitProgram(self,ctx:Program,o:object): pass

    def visitClassDecl(self,ctx:ClassDecl,o:object):pass

    def visitVarDecl(self,ctx:VarDecl,o:object):pass

    def visitFuncDecl(self,ctx:FuncDecl,o:object):pass

    def visitIntType(self,ctx:IntType,o:object):pass

    def visitFloatType(self,ctx:FloatType,o:object):pass

    def visitClassType(self,ctx:ClassType,o:object):pass

    def visitIntLit(self,ctx:IntLit,o:object):pass

    def visitId(self,ctx:Id,o:object):pass

    def visitFieldAccess(self,ctx:FieldAccess,o:object):pass

Your code starts at line 65

For example:

Test	Result
`Program([ClassDecl("x","",[FuncDecl("foo",[],([VarDecl("m",ClassType("x"))],[FieldAccess(Id("m"),"a"),FieldAccess(Id("m"),"b")])),VarDecl("a",IntType())])])`	`Undeclared Field: b`

Answer: (penalty regime: 0 %)

```
54
55      def visitIntType(self,ctx:IntType,o:object):pass
56
57      def visitFloatType(self,ctx:FloatType,o:object):pass
58
59      def visitClassType(self,ctx:ClassType,o:object):pass
60
61      def visitIntLit(self,ctx:IntLit,o:object):pass
```

```
62
63      def visitId(self,ctx:Id,o:object):
64          if ctx.name in o.get('classes').get(o['class']).get('methods').get(o['method']):
65              return o.get('classes').get(o['class']).get('methods').get(o['method']).get(ctx.name).name
66          raise UndeclaredIdentifier(ctx.name)
67
68      def visitFieldAccess(self,ctx:FieldAccess,o:object):
69          curClass = self.visit(ctx.exp, o)
70          while curClass != '':
71              fields = o.get('classes').get(curClass).get('fields')
72              if fields.get(ctx.field) != None:
73                  return fields.get(ctx.field)
74              curClass = o.get('classes').get(curClass).get('parent')
75
76          raise UndeclaredField(ctx.field)
```

	Test	Expected	Got	
✔	`Program([ClassDecl("x","",[FuncDecl("foo",[],([VarDecl("m",ClassType("x"))],[FieldAccess(Id("m"),"a"),FieldAccess(Id("m"),"b")])),VarDecl("a",IntType())])])`	`Undeclared Field: b`	`Undeclared Field: b`	✔
✔	`Program([ClassDecl("y","",[VarDecl("a",IntType()),FuncDecl("foo",[],([VarDecl("m",ClassType("y"))],[FieldAccess(Id("m"),"a")]))]),ClassDecl("x","",[FuncDecl("foo",[],([VarDecl("m",ClassType("x"))],[FieldAccess(Id("m"),"a"),FieldAccess(Id("m"),"b")])),VarDecl("a",IntType())])])`	`Undeclared Field: b`	`Undeclared Field: b`	✔

Passed all tests! ✔

Chính xác

Điểm cho bài nộp này: 1,00/1,00.

Câu hỏi **3**

Chính xác

Điểm 1,00 của 1,00

Let AST of a programming language be defined as follows:

class Program: #decl:List[ClassDecl]

class Decl(ABC): #abstract class

class ClassDecl:#name:str,parent:str,mem:List[Decl]

class VarDecl(Decl): #name:str,typ:Type

class FuncDecl(Decl): #name:str,param:List[VarDecl],body:Tuple(List[Decl],List[Expr])

class Type(ABC): #abstract class

class IntType(Type)

class FloatType(Type)

class ClassType(Type):#name:str

class Expr(ABC): #abstract class

class Lit(Expr): #abstract class

class IntLit(Lit): #val:int

class Id(Expr): #name:str

class FieldAccess(Expr): #exp:Expr,field:str

and exceptions:

class UndeclaredIdentifier(Exception): #name:str

class UndeclaredField(Exception): #name:str

Implement the methods of the following class Visitor to travel on the above AST to detect undeclared declarations (throw the exception UndeclaredIdentifier). Note that the redeclared declarations exception also is thrown if a redeclared declaration is detected:

class StaticCheck(Visitor):

```
    def visitProgram(self,ctx:Program,o:object): pass
    def visitClassDecl(self,ctx:ClassDecl,o:object):pass
    def visitVarDecl(self,ctx:VarDecl,o:object):pass
    def visitFuncDecl(self,ctx:FuncDecl,o:object):pass
    def visitIntType(self,ctx:IntType,o:object):pass
    def visitFloatType(self,ctx:FloatType,o:object):pass
    def visitClassType(self,ctx:ClassType,o:object):pass
    def visitIntLit(self,ctx:IntLit,o:object):pass
    def visitId(self,ctx:Id,o:object):pass
```


`Program([ClassDecl("x","",[VarDecl("a",IntType()),FuncDecl("foo",[],([VarDecl("m",ClassType("x"))],[FieldAccess(Id("m"),"a"),FieldAccess(Id("m"),"b")]))])])`	`Undeclared Field: b`

Answer: (penalty regime: 0 %)

```
        for x in ctx.decl:
            o[x.name] = {}
            self.visit(x, o[x.name])
        return o

    def visitClassDecl(self,ctx:ClassDecl,o:object):
        o['parent'] = ctx.parent # need update here to ref to the parent class
        o['fields'] = {}
        o['methods'] = {}
```

```
        for x in ctx.mem:
            self.visit(x, o)

    def visitVarDecl(self,ctx:VarDecl,o:object):
        if o.get('fields').get(ctx.name) != None:
            RedeclaredField(ctx.name)
        o['fields'][ctx.name] = ctx.typ

    def visitFuncDecl(self,ctx:FuncDecl,o:object):
        if o.get('methods').get(ctx.name) != None:
            RedeclaredMethod(ctx.name)
        o['methods'][ctx.name] = {}
        for x in ctx.param:
            if o.get('methods').get(ctx.name).get(x.name) != None:
                raise RedeclaredField(x.name)
```

	Test	Expected	Got	

✔	`Program([ClassDecl("x","",[VarDecl("a",IntType()),FuncDecl("foo",[],`
	`([VarDecl("m",ClassType("x"))],`
	`[FieldAccess(Id("m"),"a"),FieldAccess(Id("m"),"b")]))])])`
		`Undeclared Field: b`	`Undeclared Field: b`	✔

✔	`Program([ClassDecl("y","",[VarDecl("a",IntType())]),ClassDecl("x","y",`
	`[FuncDecl("foo",[],([VarDecl("m",ClassType("x"))],`
	`[FieldAccess(Id("m"),"a"),FieldAccess(Id("m"),"b")]))])])`
		`Undeclared Field: b`	`Undeclared Field: b`	✔

✔	`Program([ClassDecl("y","",[VarDecl("a",IntType())]),`
	`\t ClassDecl("z","",[VarDecl("b",IntType())]),`
	`\t ClassDecl("t","y",[VarDecl("b",IntType())]),`
	`\t ClassDecl("x","y",[FuncDecl("foo",[],`
	`([VarDecl("m",ClassType("x"))],`
	`[FieldAccess(Id("m"),"a"),FieldAccess(Id("m"),"b")]))])])`
		`Undeclared Field: b`	`Undeclared Field: b`	✔

Passed all tests! ✔

Chính xác

Điểm cho bài nộp này: 1,00/1,00.

◄ Sequence Quiz

Chuyển tới...

Link Video ngày 12/10/2021 ►