Đã bắt đầu vào lúc	Tuesday, 21 September 2021, 8:02 AM
Tình trạng	Đã hoàn thành
Hoàn thành vào lúc	Monday, 27 September 2021, 3:03 AM
Thời gian thực hiện	5 ngày 19 giờ
Điểm	6,00/6,00
Điểm	**10,00** của 10,00 (**100**%)

Câu hỏi **1**

Chính xác

Điểm 1,00 của 1,00

Given the grammar of MP as follows:

program: vardecls EOF;

vardecls: vardecl vardecltail;

vardecltail: vardecl vardecltail | ;

vardecl: mptype ids ';' ;

mptype: INTTYPE | FLOATTYPE;

ids: ID ',' ids | ID;

INTTYPE: 'int';

FLOATTYPE: 'float';

ID: [a-z]+ ;

Please copy the following class into your answer and modify the bodies of its methods to count the terminal nodes in the parse tree? Your code starts at line 10.

```
class TerminalCount(MPVisitor):
    def visitProgram(self,ctx:MPParser.ProgramContext):
        return None
    def visitVardecls(self,ctx:MPParser.VardeclsContext):
        return None
    def visitVardecltail(self,ctx:MPParser.VardecltailContext):
        return None
    def visitVardecl(self,ctx:MPParser.VardeclContext):
        return None
    def visitMptype(self,ctx:MPParser.MptypeContext):
        return None
    def visitIds(self,ctx:MPParser.IdsContext):
        return None
```

Answer: (penalty regime: 0 %)

Ace editor not ready. Perhaps reload page?
Falling back to raw text area.

```
class TerminalCount(MPVisitor):

    def visitProgram(self,ctx:MPParser.ProgramContext):

        return 1 + self.visitVardecls(ctx.vardecls())


    def visitVardecls(self,ctx:MPParser.VardeclsContext):

        return self.visitVardecl(ctx.vardecl()) + self.visitVardecltail(ctx.vardecltail())

    def visitVardecltail(self,ctx:MPParser.VardecltailContext):

        return self.visitVardecl(ctx.vardecl()) + self.visitVardecltail(ctx.vardecltail()) if
ctx.vardecl() else 0

    def visitVardecl(self,ctx:MPParser.VardeclContext):
        return 1 + self.visitMptype(ctx.mptype()) + self.visitIds(ctx.ids())
```

	Test	Expected	Got	
✔	`"int a;"`	4	4	✔
✔	`"""int a,b;"""`	6	6	✔
✔	`"int a;float b;"`	7	7	✔
✔	`"int a,b;float c;"`	9	9	✔
✔	`"int a,b;float c,d,e;"`	13	13	✔

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

Given the grammar of MP as follows:

program: vardecls EOF;

vardecls: vardecl vardecltail;

vardecltail: vardecl vardecltail | ;

vardecl: mptype ids ';' ;

mptype: INTTYPE | FLOATTYPE;

ids: ID ',' ids | ID;

INTTYPE: 'int';

FLOATTYPE: 'float';

ID: [a-z]+ ;

Please copy the following class into your answer and modify the bodies of its methods to return the height of the parse tree? Your code starts at line 10.

```
class TerminalCount(MPVisitor):
    def visitProgram(self,ctx:MPParser.ProgramContext):
        return None
    def visitVardecls(self,ctx:MPParser.VardeclsContext):
        return None
    def visitVardecltail(self,ctx:MPParser.VardecltailContext):
        return None
    def visitVardecl(self,ctx:MPParser.VardeclContext):
        return None
    def visitMptype(self,ctx:MPParser.MptypeContext):
        return None
    def visitIds(self,ctx:MPParser.IdsContext):
        return None
```

Answer: (penalty regime: 0 %)

Ace editor not ready. Perhaps reload page?
Falling back to raw text area.

```
class TerminalCount(MPVisitor):

    def visitProgram(self, ctx: MPParser.ProgramContext):
        if(self.visitVardecls(ctx.vardecls())):
            return 1+self.visitVardecls(ctx.vardecls())
        return 1

    def visitVardecls(self, ctx: MPParser.VardeclsContext):
        if (self.visitVardecl(ctx.vardecl()) >= self.visitVardecltail(ctx.vardecltail())):
            return self.visitVardecl(ctx.vardecl())+1
        return self.visitVardecltail(ctx.vardecltail())+1

    def visitVardecltail(self, ctx: MPParser.VardecltailContext):
        if (ctx.getChildCount() == 2 and self.visitVardecl(ctx.vardecl()) >=
self.visitVardecltail(ctx.vardecltail())):
            return self.visitVardecl(ctx.vardecl())+1
        if (ctx.getChildCount() == 2 and self.visitVardecl(ctx.vardecl()) <
self.visitVardecltail(ctx.vardecltail())):
```

	Test	Expected	Got	
✔	"int a;"	4	4	✔
✔	"""int a,b;"""	5	5	✔
✔	"int a;float b;"	5	5	✔
✔	"int a,b;float c;"	5	5	✔
✔	"int a,b;float c,d,e;"	7	7	✔

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

Given the grammar of MP as follows:

program: exp EOF;

exp: term ASSIGN exp | term;

term: factor COMPARE factor | factor;

factor: factor ANDOR operand | operand;

operand: ID | INTLIT | BOOLIT | '(' exp ')';

INTLIT: [0-9]+ ;

BOOLIT: 'True' | 'False' ;

ANDOR: 'and' | 'or' ;

ASSIGN: '+=' | '-=' | '&=' | '|=' | ':=' ;

COMPARE: '=' | '<>' | '>=' | '<=' | '<' | '>' ;

ID: [a-z]+ ;

and AST classes as follows:

class Expr(ABC):

class Binary(Expr): #op:string;left:Expr;right:Expr

class Id(Expr): #value:string

class IntLiteral(Expr): #value:int

class BooleanLiteral(Expr): #value:boolean

Please copy the following class into your answer and modify the bodies of its methods to generate the AST of a MP input?

```
class ASTGeneration(MPVisitor):
    def visitProgram(self,ctx:MPParser.ProgramContext):
        return None
    def visitExp(self,ctx:MPParser.ExpContext):
        return None
    def visitTerm(self,ctx:MPParser.TermContext):
        return None
    def visitFactor(self,ctx:MPParser.FactorContext):
        return None
    def visitOperand(self,ctx:MPParser.OperandContext):
        return None
```

Answer: (penalty regime: 0 %)

Ace editor not ready. Perhaps reload page?
Falling back to raw text area.

```
class ASTGeneration(MPVisitor):

    def visitProgram(self, ctx: MPParser.ProgramContext):
        return self.visit(ctx.exp())

    def visitExp(self, ctx: MPParser.ExpContext):
        if ctx.getChildCount() == 3:
            return Binary(ctx.ASSIGN().getText(), self.visitTerm(ctx.term()),
self.visitExp(ctx.exp()))
        return self.visit(ctx.term())

    def visitTerm(self, ctx: MPParser.TermContext):
        if ctx.getChildCount() == 3:
            return Binary(ctx.COMPARE().getText(), self.visitFactor(ctx.factor(0)),
self.visitFactor(ctx.factor(1)))
        return self.visit(ctx.factor(0))

    def visitFactor(self, ctx: MPParser.FactorContext):
```

	Test	Expected	Got
✔	`"a := b := 4"`	`Binary(:=,Id(a),Binary(:=,Id(b),IntLiteral(4)))`	`Binary(:=,Id(a),Binary(:=,Id(b),IntLiteral(4)))`
✔	`"""a += b -= a and (b > 3)"""`	`Binary(+=,Id(a),Binary(-=,Id(b),Binary(and,Id(a),Binary(>,Id(b),IntLiteral(3)))))`	`Binary(+=,Id(a),Binary(-=,Id(b),Binary(and,Id(a),Binary(>,Id(b),IntLiter`
✔	`"a or b and True"`	`Binary(and,Binary(or,Id(a),Id(b)),BooleanLiteral(True))`	`Binary(and,Binary(or,Id(a),Id(b)),BooleanLiteral`

Passed all tests! ✔

Chính xác

Điểm cho bài nộp này: 1,00/1,00.

Câu hỏi **4**

Chính xác

Điểm 1,00 của 1,00

Given the grammar of MP as follows:

program: vardecls EOF;

vardecls: vardecl vardecltail;

vardecltail: vardecl vardecltail | ;

vardecl: mptype ids ';' ;

mptype: INTTYPE | FLOATTYPE;

ids: ID ',' ids | ID;

INTTYPE: 'int';

FLOATTYPE: 'float';

ID: [a-z]+ ;

and AST classes as follows:

class Program:#decl:list(VarDecl)

class Type(ABC): pass

class IntType(Type): pass

class FloatType(Type): pass

class VarDecl: #variable:Id; varType: Type

class Id: #name:str

Please copy the following class into your answer and modify the bodies of its methods to generate the AST of a MP input?

```
class ASTGeneration(MPVisitor):
    def visitProgram(self,ctx:MPParser.ProgramContext):
        return None
    def visitVardecls(self,ctx:MPParser.VardeclsContext):
        return None
    def visitVardecltail(self,ctx:MPParser.VardecltailContext):
        return None
    def visitVardecl(self,ctx:MPParser.VardeclContext):
        return None
    def visitMptype(self,ctx:MPParser.MptypeContext):
        return None
    def visitIds(self,ctx:MPParser.IdsContext):
        return None
```

Answer: (penalty regime: 0 %)

Ace editor not ready. Perhaps reload page?
Falling back to raw text area.

```
class ASTGeneration(MPVisitor):
    def visitProgram(self,ctx:MPParser.ProgramContext):
        vardecls = self.visit(ctx.vardecls())
        return Program(vardecls)

    def visitVardecls(self,ctx:MPParser.VardeclsContext):
        vardecl = self.visit(ctx.vardecl())
        vardecltail = self.visit(ctx.vardecltail())
        if vardecltail is None:
            return vardecl
        return vardecl + vardecltail

    def visitVardecltail(self,ctx:MPParser.VardecltailContext):
        if ctx.vardecltail() is None:
            return None
        vardecl = self.visit(ctx.vardecl())
        vardecltail = self.visit(ctx.vardecltail())
        if vardecltail is None:
```

	Test	Expected
✔	`"int a;"`	`Program([VarDecl(Id(a),IntType)])`
✔	`"""int a,b;"""`	`Program([VarDecl(Id(a),IntType),VarDecl(Id(b),IntType)])`
✔	`"int a;float b;"`	`Program([VarDecl(Id(a),IntType),VarDecl(Id(b),FloatType)])`
✔	`"int a,b;float c;"`	`Program([VarDecl(Id(a),IntType),VarDecl(Id(b),IntType),VarDecl(Id(c),FloatType)])`
✔	`"int a,b;float c,d,e;"`	`Program([VarDecl(Id(a),IntType),VarDecl(Id(b),IntType),VarDecl(Id(c),FloatType),VarDecl(Id(d),FloatType`

Passed all tests! ✔

Chính xác

Điểm cho bài nộp này: 1,00/1,00.

Câu hỏi **5**

Chính xác

Điểm 1,00 của 1,00

Given the grammar of MP as follows:

program: vardecl+ EOF;

vardecl: mptype ids ';' ;

mptype: INTTYPE | FLOATTYPE;

ids: ID (',' ID)*;

INTTYPE: 'int';

FLOATTYPE: 'float';

ID: [a-z]+ ;

and AST classes as follows:

class Program:#decl:list(VarDecl)

class Type(ABC): pass

class IntType(Type): pass

class FloatType(Type): pass

class VarDecl: #variable:Id; varType: Type

class Id: #name:str

Please copy the following class into your answer and modify the bodies of its methods to generate the AST of a MP input?

```
class ASTGeneration(MPVisitor):
    def visitProgram(self,ctx:MPParser.ProgramContext):
        return None
    def visitVardecl(self,ctx:MPParser.VardeclContext):
        return None
    def visitMptype(self,ctx:MPParser.MptypeContext):
        return None
    def visitIds(self,ctx:MPParser.IdsContext):
        return None
```

Answer: (penalty regime: 0 %)

Ace editor not ready. Perhaps reload page?

Falling back to raw text area.

```
class ASTGeneration(MPVisitor):

    def visitProgram(self,ctx:MPParser.ProgramContext):

        body = []
        for decl in ctx.vardecl():
            body = body + self.visit(decl)
        return Program(body)

    def visitVardecl(self,ctx:MPParser.VardeclContext):
        typ = self.visit(ctx.mptype())
        ids = self.visit(ctx.ids())
        list_var = []
        for i in ids:
            list_var.append(VarDecl(i,typ))
        return list_var
```

	Test	Expected
✔	`"int a;"`	`Program([VarDecl(Id(a),IntType)])`
✔	`"""int a,b;"""`	`Program([VarDecl(Id(a),IntType),VarDecl(Id(b),IntType)])`
✔	`"int a;float b;"`	`Program([VarDecl(Id(a),IntType),VarDecl(Id(b),FloatType)])`
✔	`"int a,b;float c;"`	`Program([VarDecl(Id(a),IntType),VarDecl(Id(b),IntType),VarDecl(Id(c),FloatType)])`
✔	`"int a,b;float`	`Program([VarDecl(Id(a),IntType),VarDecl(Id(b),IntType),VarDecl(Id(c),FloatType),VarDecl(Id(d),FloatType`

Câu hỏi **6**

Chính xác

Điểm 1,00 của 1,00

Given the grammar of MP as follows:

program: exp EOF;

exp: (term ASSIGN)* term;

term: factor COMPARE factor | factor;

factor: operand (ANDOR operand)*;

operand: ID | INTLIT | BOOLIT | '(' exp ')';

INTLIT: [0-9]+ ;

BOOLIT: 'True' | 'False' ;

ANDOR: 'and' | 'or' ;

ASSIGN: '+=' | '-=' | '&=' | '|=' | ':=' ;

COMPARE: '=' | '<>' | '>=' | '<=' | '<' | '>' ;

ID: [a-z]+ ;

and AST classes as follows:

class Expr(ABC):

class Binary(Expr): #op:string;left:Expr;right:Expr

class Id(Expr): #value:string

class IntLiteral(Expr): #value:int

class BooleanLiteral(Expr): #value:boolean

Please copy the following class into your answer and modify the bodies of its methods to generate the AST of a MP input?

```
class ASTGeneration(MPVisitor):
    def visitProgram(self,ctx:MPParser.ProgramContext):
        return None
    def visitExp(self,ctx:MPParser.ExpContext):
        return None
    def visitTerm(self,ctx:MPParser.TermContext):
        return None
    def visitFactor(self,ctx:MPParser.FactorContext):
        return None
    def visitOperand(self,ctx:MPParser.OperandContext):
        return None
```

Answer: (penalty regime: 0 %)

Ace editor not ready. Perhaps reload page?
Falling back to raw text area.

```
from functools import reduce
class ASTGeneration(MPVisitor):

    def visitProgram(self, ctx: MPParser.ProgramContext):

        return self.visit(ctx.exp())

    def visitExp(self, ctx: MPParser.ExpContext):
        if len(ctx.ASSIGN()) == 0: return self.visit(ctx.term(0))
        assigns = ctx.ASSIGN()
        terms = ctx.term()
        lst = list(zip(assigns, terms[:-1]))[::-1] # zip and reverse
        return reduce(lambda x,y: Binary(y[0].getText(), self.visit(y[1]), x), lst,
self.visit(terms[-1]))


    def visitTerm(self, ctx: MPParser.TermContext):
        if ctx.getChildCount() == 3:
```

	Test	Expected	Got
✔	`"a := b := 4"`	`Binary(:=,Id(a),Binary(:=,Id(b),IntLiteral(4)))`	`Binary(:=,Id(a),Binary(:=,Id(b),IntLiteral(4)))`
✔	`"""a += b -= a and (b > 3)"""`	`Binary(+=,Id(a),Binary(-=,Id(b),Binary(and,Id(a),Binary(>,Id(b),IntLiteral(3)))))`	`Binary(+=,Id(a),Binary(-=,Id(b),Binary(and,Id(a),Binary(>,Id(b),IntLiter`
✔	`"a or b and True"`	`Binary(and,Binary(or,Id(a),Id(b)),BooleanLiteral(True))`	`Binary(and,Binary(or,Id(a),Id(b)),BooleanLiteral`

Passed all tests! ✔

Chính xác

Điểm cho bài nộp này: 1,00/1,00.

◄ AST Quiz

Chuyển tới...

Link Video của buổi học 21/9/2021 ►