

**<program>** -> <dec> <main> <dec>

**<main>** -> main ( ) { <mst> }

**<dec>** -> <class\_dec> | <fn\_dec> | ε

**<class\_dec>** -> class ID <extends> { <class\_body> }

**<extends>** -> extends:ID | null

**<class\_body>** -> <class\_chidlren> <class\_body> | null

**<class\_chidlren>** -> <dt\_dec> | <func\_dec> | <Constructor>

**<func\_dec>** -> DT ID ( <param\_list> ) <Body>

**<param\_list>** -> <param> | <param> , <param\_list> | ε

**<param>** -> DT ID

**<Body>** -> ; | { <MST> }

**<Constructor>** -> ID ( <param\_list> ) <Body>

**<dt\_dec>** -> DT ID = <Const\_or\_ID> ; // int x = 5; or int x = x;  
| DT ID ; // int x;  
| DT <multi\_dec> ; // int x,b,c; or int a,b;  
| DT <multi\_dec\_init> ; // int x = 10,b=20; or int x = a, y = b; or int x=a,y=3;  
| <arr\_type> ID [] = { <arr\_const\_or\_id> }; // int arr[] = {}; or int arr[] = {1,2,a}; obj arr[]

**<Const\_or\_ID>** -> Const | ID

**<multi\_dec>** -> ID , <multi\_dec> | ID

**<multi\_dec\_init>** -> ID = Const, <multi\_dec\_init> | ID = ID, <multi\_dec\_init> | ID = <Const\_or\_ID>

**<arr\_type>** -> DT | ID

**<arr\_const\_or\_id>** -> ε | <Const\_or\_ID> | ID , | Const ,

**<SST>** -> <while\_loop> | <for\_loop> | <if\_else> | <do\_while> | <command> | <exp> | <try> |

<throw> | <return> | <continue> | <break> | <dt\_dec> | <func\_dec>

**<MST>** -> <SST><MST> | ε

**<while\_loop>** -> while (<cond>)<loop\_body>

**<cond>** -> <Const\_or\_ID> | <Const\_or\_ID> <ROP> <Const\_or\_ID> | <exp>

**<ROP>** -> RO1 | RO2

**<loop\_body>** -> ; | <SST> | {<MST>}

**<for\_loop>** -> for (<F1><F2>;<F3>) <loop\_body>

**<F1>** -> <dt\_dec> | <assign\_st> | ;

**<F2>** -> <cond> | null

**<F3>** -> <inc\_dec> | <assign\_st> | null

**<inc\_dec>** -> ID increase\_decrease  
**<assign\_st>** -> ID = <assign\_options> ;  
**<assign\_options>** -> <Const\_or\_ID> | <exp>

**<if>** -> if (<cond>) <loop\_body> <else>  
**<else>** -> else <loop\_body> | null

**<do\_while>** -> do <loop\_body> while (<cond>);

**<this>** -> this ID ; | this <func\_call> ;  
**<func\_call>** -> ID ( <param\_list> )

**<exp>** -> <OE>  
**<OE>** -> <AE><OE'>  
**<OE'>** -> OR<AE><OE'> | null  
**<AE>** -> <RE2><AE'>  
**<AE'>** -> AND<RE2><AE'> | null  
**<RE2>** -> <RE1><RE2'>  
**<RE2'>** -> RO2<RE1><RE2'> | null  
**<RE1>** -> <E><RE1'>  
**<RE1'>** -> RO1<E><RE1'> | null  
**<E>** -> <T><E'>  
**<E'>** -> PM<T><E'> | null  
**<T>** -> <F><T'>  
**<T'>** -> MDM<F><T'> | null  
**<F>** -> ID | const | ( <OE> ) | -<F> | NOT <F>

**<return>** -> return <return\_options> ;  
**<return\_options>** -> ID | <const> | <exp> | null

**<continue>** -> continue;

**<break>** -> break;

**<try>** → try { <MST> } catch ( ID ) { <MST> }

**<throw>** -> throw <throw\_options>:  
**<throw\_options>** -> ID | Const | new ID ( <param\_list> )