CD – JAVA

Sl no.	Name	Registration number	Roll number
1.	Disha Agarwal	210905412	62
2.	Soumya Sahu	210905196	35
3.	Samarth Parashar	210905206	40
4.	Vaibhav Sandhir	210905152	24
5.	Aditya Singhvi	210905216	41

FLEX CODE

```
%{
#include<stdio.h>
#include "final.tab.h"
%}

%%
[0-9]+ {return NUM;}
[0-9]+"."[0-9]+ {return NUM;}
([*=(){}/%;]|"+"|","|"-"|".") {return yytext[0];}
"class" {return CLASS;}
"import" {return IMPORT;}
"public" {return PUBLIC;}
"private" {return PRIVATE;}
"protected" {return PROTECTED;}
"static" {return STATIC;}
"int" {return INT;}
"double" {return DOUBLE;}
"char" {return CHAR;}
"boolean" {return BOOLEAN;}
"String" {return STRING;}
"float" {return FLOAT;}
"void" {return VOID;}
"true" {return TRUE;}
"false" {return FALSE;}
\"[^"]*\" {return STRINGLIT;}
'[^']?' {return CHARLIT;}
[_$a-zA-Z][a-zA-Z0-9_$]* {return IDENTIFIER;}
%%

int yywrap()
{
  return 1;
}
```

BISON CODE

```
%{
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
int yylex();
int yyerror();
extern FILE* yyin;
%}

%name final
%token NUM IDENTIFIER NL STRINGLIT CHARLIT
%token CLASS IMPORT
%token PUBLIC PRIVATE PROTECTED STATIC
%token INT DOUBLE CHAR BOOLEAN STRING FLOAT
%token VOID TRUE FALSE
%left '+' '-'
%left '*' '/' '%'

%%

prog : importst classdecl ;

importst : IMPORT import_clause ';' importst |  ;
import_clause : IDENTIFIER '.' package_name ;
package_name : IDENTIFIER more_package_parts| '*' ;
more_package_parts : '.' package_name|  ;

classdecl : CLASS IDENTIFIER '{' classbody '}' ;
classbody : classmember classbody |  ;
classmember : accessmod stat dtype IDENTIFIER mem | accessmod stat VOID IDENTIFI
ER methoddecl;
accessmod : PUBLIC | PRIVATE | PROTECTED |  ;
stat : STATIC |  ;
dtype : INT | DOUBLE | CHAR | BOOLEAN | STRING | FLOAT ;
mem : instancevar | methoddecl ;

instancevar : initialise ';';
initialise : '=' literal |  ;
```

```
methoddecl : '(' paramlist ')' '{' statement_list '}' ;
paramlist : dtype IDENTIFIER moreparam |  ;
moreparam : ',' dtype IDENTIFIER moreparam |  ;
statement_list : statement statement_list |  ;
statement : decl ';' | assign_stat ';' ;
decl : dtype IDENTIFIER assign moreid ;
moreid : ',' IDENTIFIER assign moreid |  ;
assign : '=' val |  ;
assign_stat : IDENTIFIER '=' val ;

val : expn | STRINGLIT | TRUE | FALSE ;
expn : expn '+' term | expn '-' term | term ;
term : term '*' factor | term '/' factor | term '%' factor | factor ;
factor : NUM | IDENTIFIER | CHARLIT | '(' expn ')' ;
literal : NUM | CHARLIT | STRINGLIT | TRUE | FALSE ;
%%


int yyerror(char *msg)
{
  printf("************* %s *************\n",msg);
  exit(0);
}
void main (int argc, char *args[])
{
  yyin=fopen(args[1],"r");
  if(!yyin)
  {
    yyerror("File Error");
  }
  do
  {
    if(yyparse())
      yyerror("Parse");
  } while (!feof(yyin));
  printf("************* SUCCESS *************\n");
}
```

SAMPLE INPUTS

```
import java.util.*;  // importing all classes from java.util
class ABC  // class declaration
{
    // empty class
}
```

Example 1 - Successful

```
// multiple import statements
import java.util.*;
import java.util.Scanner;  // importing Scanner class from java.util

class Person
{
    // multiple method declarations
    private static int age;  // Access modifier + Static + Instance variable
    private String name;  // Without Static
    char section = 'A';   // Without Static and Access modifier + Initialisation

    public static void main()  // static method declaration with access modifier
    {

       int year;  // declaration statement
       year = 2023;  // assignment statement
       boolean flag = TRUE;  // declaration + assignment

    }
}
```

Example 2 - Successful

```
// no import statements
class Person
{
    // no instance variables
    // multiple method declarations
    public static void main() // method without parameters
    {
       int temp = 5 + 10;
    }
    void send(int b, char c) // method with two parameters
    {
        // empty method body
    }
}
```

Example 3 - Successful

```
class Calculate
{
    public static void main()
    {
        // declaration statement with multiple variables + assignment
        int a = 1, b = 0;
        // addition and subtraction with integer constants and variable
        a = a + 20 - 30;
        // division and multiplication with floating constants and variable
        b = 20.3 * 40.27 / a;
        // different types of operators, operands and paranthesis
        int c = (a - b + 2067.0) / (a * b + 400);
    }
}
```

Example 4 - Successful

```
class Assign
{
    public static void main()
    {
        int a, b;
        3000 = b;  // incorrect assignment statement
    }
}
```

Example 5 - Error