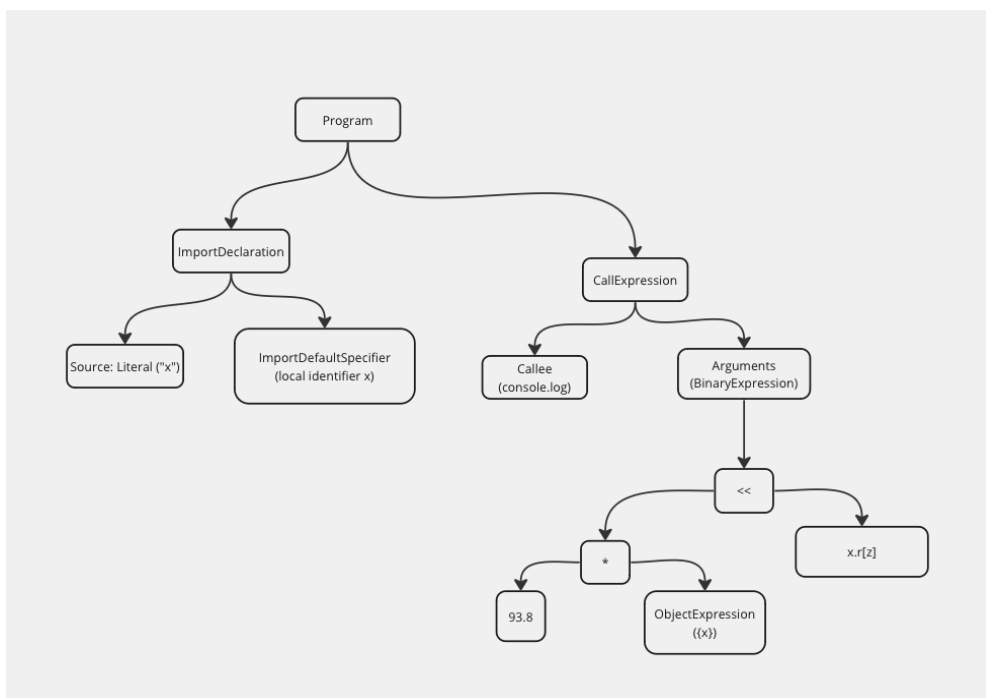
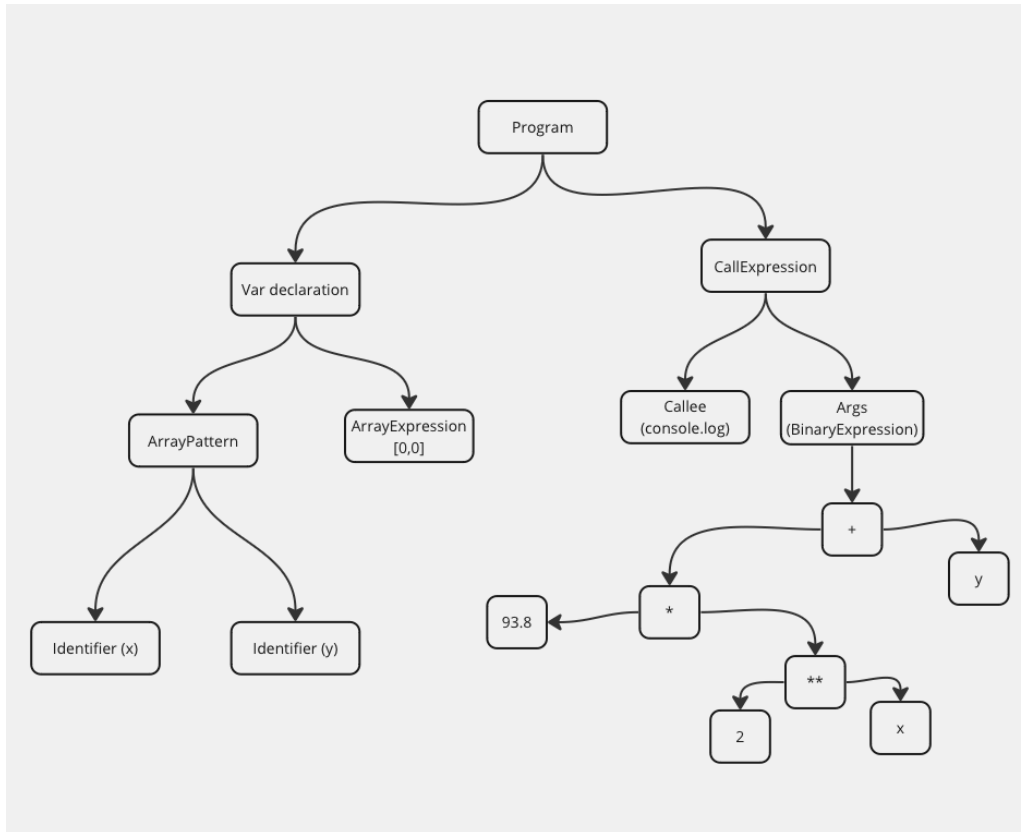
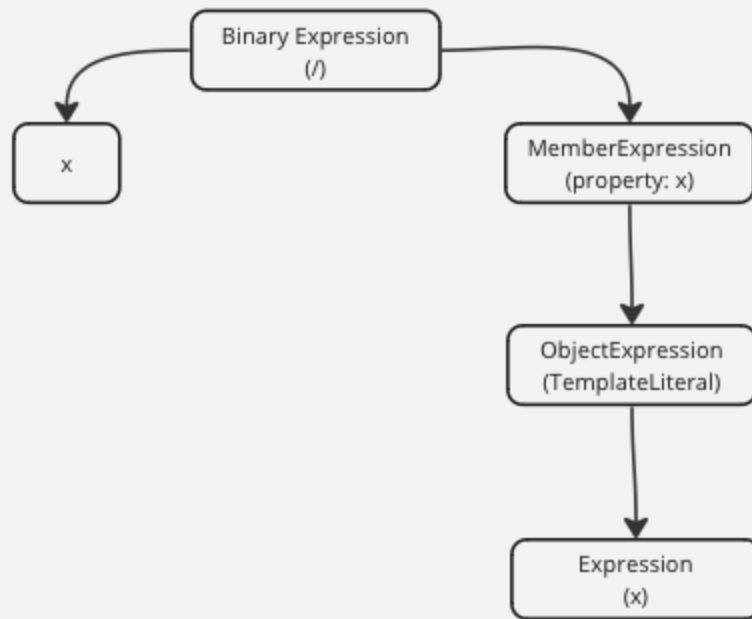
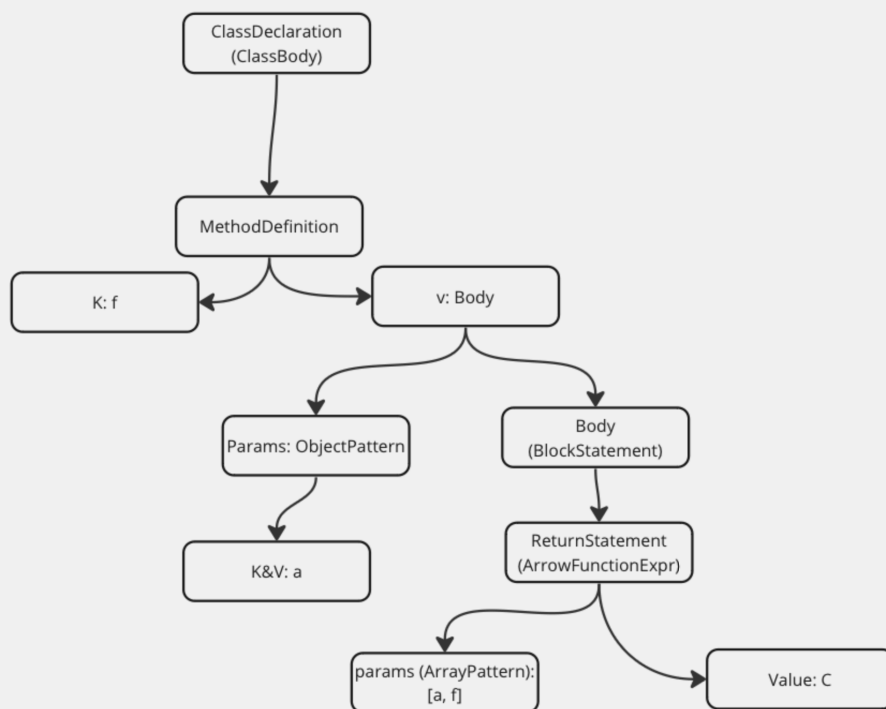


Homework 1





3.



4.

5.

```
Arithmetic {  
  Exp = PowExp  
  PriExp  
    = "-" PriExp -- neg  
    | number  
  PowExp  
    = PriExp "**" PowExp -- pow  
    | PriExp  
  number = digit+  
}
```

a.

```
Arithmetic {  
  Exp = PriExp  
  PriExp  
    = "-" PriExp -- neg  
    | PowExp  
  PowExp  
    = PriExp "**" PowExp -- pow  
    | number  
  number = digit+  
}
```

b.

```
Arithmetic {  
  Exp = PowExp  
  PowExp  
    = PriExp "**" PowExp -- pow  
    | PriExp  
    | NegExp  
  PriExp  
    = "(" Exp ")" -- paren  
    | number  
  NegExp  
    = "-" PriExp -- neg  
    | PriExp  
  number = digit+  
}
```

c.

```

Astro {
  Program      = Statement+
  Statement    = id "=" Exp ";"           --assignment
                | print Exp ";"           --print
                | "if" "(" Exp ")" Block ElsePart --if
                | "while" "(" Exp ")" Block --while
  ElsePart     = "else" (Block | Statement_if) --else
  Block        = "{" Statement* "}"
  Exp          = Exp ("+" | "-") Term      --binary
                | Term
                | RelExp
  RelExp       = Term ("<=" | ">=" | "<" | ">" | "==" | "!=") Term --relational
  Term         = Term ("*" | "/" | "%") Factor --binary
                | Factor
  Factor       = Primary "**" Factor        --binary
                | "-" Primary             --negation
                | Primary
  Primary      = id "(" ListOf<Exp, ",", ">" --call
                | numeral                 --num
                | id                       --id
                | "(" Exp ")"              --parens
  numeral      = digit+ ( "." digit+ )? ( ("E" | "e") ("+" | "-")? digit+ )?
  print        = "print" ~idchar
  idchar       = letter | digit | "_"
  id           = ~print letter idchar*
  space        += "//" ( ~"\n" any )*      --comment
}

```

6.