Grammar Subset for Assignment 2

Concrete syntax

<expr> ::=   <conditional_expr> | <or_expr>

<conditional_expr> ::= **if** <expr> **?** <expr> **?** <expr>

<or_expr> ::= <and_expr> ((**|** | **||**) <and_expr>)*

<and_expr> ::= <comparison_expr> ((**&** | **&&**) <comparison_expr>)*

<comparison_expr> ::=   <power_expr> ((**<** | **>** | **==** | **<=** | **>=**) <power_expr>)*

<power_expr> ::=   <additive_expr> ****** <power_expr> | <additive_expr>

<additive_expr> ::= <multiplicative_expr> ((**+** | **-**) <multiplicative_expr>)*

<multiplicative_expr> ::= <unary_expr> ((***** | **/** | **%**) <unary_expr>)*

<unary_expr> ::= (**!** | **-** | **sin** | **cos** | **atan**) <unary_expr> | <primary_expr>

<primary_expr> ::=
  STRING_LIT |
  NUM_LIT |
  IDENT |
  (<expr> **)** |
  Z |
  rand

Abstract Syntax

Expr ::= ConditionalExpr | BinaryExpr | UnaryExpr | StringLitExpr | IdentExpr
      NumLitExpr | ZExpr | RandExpr
ConditionalExpr ::= Expr Expr Expr
BinaryExpr ::= Expr op Expr
UnaryExpr ::= op Expr