**Grammar**

start symbol:Program

**reserved words**:  
Identifier  
Value  
Condition

|  |  |  |
| --- | --- | --- |
| Program | ::= StatementCol |  |
| StatementCol | ::= StatementCol Statement  | Statement |  |
| Statement | ::= Declaration  | Assignment  | ControlFlow |  |
| Block | ::= **:** StatementCol **..**  | **:** StatementCol **>> Identifier;..** |  |
| Type (bool, number, string) | ::= **½**  | **#**  | **§** |  |
| Declaration | ::=Type **Identifier;**  | Type **Identifier** Assignment  | **Identifier-**ParamList**->**Type Block |  |
| Assignment | ::= **Identifier** = **Value;**  | **Identifier = Identifier;** |  |
| ParamList | ::= Type **Identifier** ParamListTail  | ε |  |
| ParamListTail | ::= **,** Type **Identifier** ParamListTail  | ε |  |
| ControlFlow (if,  if else,  while) | ::= **? Condition** Block  | **? Condition** Block **¤** Block  | **! Condition** Block |  |
| FunctionCall | ::= **Indentifier-** ArgumentList **-;** |  |
| ArgumentList | ::= **Identifier** ArgumentListTail  | **Value**  | ε |  |
| ArgumentListTail | ::= **, Identifier** ArgumentListTail  | **, Value** ArgumentListTail  | ε |  |
| ConsoleOut | ::= **Identifier >>;** |  |
| ConsoleIn | ::= **Identifier <<;** |  |

**V2:**

|  |  |
| --- | --- |
| Program | ::= StatementCol |
| StatementCol | ::= (Statement)\* |
| Statement | ::= Declaration  | Assignment  | ControlFlow  | ConsoleOut  | ConsoleIn |
| Block | ::= **:** StatementCol (**->** MathExpr**;**) **..** |
| Type (bool, number, string) | ::= **½**  | **#**  | **§** |
| Declaration | ::=Type **Identifier** (**=** MathExpr);  | **<->Identifier(**ParamList**)**Type Block |
| Assignment | ::= **Identifier** **=** MathExpr**;** |
| ParamList | ::= Type **Identifier** ParamListTail  | ε |
| ParamListTail | ::= **,** Type **Identifier** ParamListTail  | ε |
| ControlFlow (if,  if else,  while) | ::= **? Condition** Block  | **? Condition** Block **¤** Block  | **! Condition** Block |
| FunctionArguments | ::= **(** ArgumentList **);** |
| ArgumentList | ::= MathExpr ArgumentListTail  | ε |
| ArgumentListTail | ::= **,** MathExpr ArgumentListTail  | ε |
| MathExpr | ::= SecondMathExpr  | MathExpr AddSubOp SecondMathExpr |
| SecondMathExpr | ::= PrimaryMathExpr  | SecondMathExpr MulDivOP PrimaryMathExpr |
| PrimaryMathExpr | ::= **Value**  | **Identifier** ( FuctionArguments )  | **(** MathExpr **)** |
| AddSubOp | ::= **+**  | **-** |
| MulDivOP | ::= **\***  | **/** |
| ConsoleOut | ::= >> MathExpr**;** |
| ConsoleIn | ::= << MathExpr**;** |

**V3:**

|  |  |
| --- | --- |
| Program | ::= StatementCol |
| StatementCol | ::= (Statement)\* |
| Statement | ::= Declaration  | Assignment  | ControlFlow  | ConsoleOut  | ConsoleIn |
| Block | ::= **:** StatementCol (**->** MathExpr**;**) **..** |
| Type (number, string) | ::= **#**  | **§** |
| Declaration | ::=Type **Identifier** (**=** MathExpr)**;**  | Type **Identifier[**(**Numaric**)**]** (**= {**ArgumentList**}**)**;**  | **<->Identifier(**ParamList**)**Type Block |
| Assignment | ::= **Identifier** **=** MathExpr**;**  | **Identifier[Numeric] =** ArgumentList**;** |
| ParamList | ::= Type **Identifier** ParamListTail  | ε |
| ParamListTail | ::= **,** Type **Identifier** ParamListTail  | ε |
| ControlFlow (if,  if else,  while) | ::= **? Condition** Block  | **? Condition** Block (**¤** Block)  | **! Condition** Block |
| FunctionArguments | ::= **(** ArgumentList **);** |
| ArgumentList | ::= MathExpr ArgumentListTail  | ε |
| ArgumentListTail | ::= **,** MathExpr ArgumentListTail  | ε |
| MathExpr | ::= SecondMathExpr  | MathExpr AddSubOp SecondMathExpr |
| SecondMathExpr | ::= PrimaryMathExpr  | SecondMathExpr MulDivOP PrimaryMathExpr |
| PrimaryMathExpr | ::= **Value**  | **Identifier** ( FuctionArguments )  | **(** MathExpr **)** |
| AddSubOp | ::= **+**  | **-** |
| MulDivOP | ::= **\***  | **/** |
| Comparison | ::= <  | >  | == |
| Comparison | ::= MathExpr Comparison MathExpr |
| ConsoleOut | ::= >> MathExpr**;** |
| ConsoleIn | ::= << MathExpr**;** |

**V3 - fixed:**

|  |  |
| --- | --- |
| Program | ::= StatementCol |
| StatementCol | ::= (Statement)\* |
| Statement | ::= Declaration  | Assignment  | ControlFlow  | ConsoleOut  | ConsoleIn |
| Block | ::= **:** StatementCol ( **->** MathExpr**;** | ε ) **..** |
| Type (number, string) | ::= **#**  | **§** |
| Declaration | ::=Type **Identifier** ( **=** MathExpr | ε )**;**  | Type **Identifier[**( **Numaric** |ε )**]** ( **= {**ArgumentList**}** |ε )**;**  | **<->Identifier(**ParamList**)**Type Block |
| Assignment | ::= **Identifier**( **[Numeric]** | ε ) **=** MathExpr**;** |
| ParamList | ::= Type **Identifier** ParamListTail  | ε |
| ParamListTail | ::= **,** Type **Identifier** ParamListTail  | ε |
| ControlFlow (if,  if else,  while) | ::= **? Condition** Block ( **¤** Block | ε )  | **! Condition** Block |
| FunctionArguments | ::= **(** ArgumentList **);** |
| ArgumentList | ::= MathExpr ArgumentListTail  | ε |
| ArgumentListTail | ::= **,** MathExpr ArgumentListTail  | ε |
| MathExpr | ::= SecondMathExpr  | MathExpr AddSubOp SecondMathExpr |
| SecondMathExpr | ::= PrimaryMathExpr  | SecondMathExpr MulDivOP PrimaryMathExpr |
| PrimaryMathExpr | ::= **Value**  | **Identifier** ( FuctionArguments | ε )  | **(** MathExpr **)** |
| AddSubOp | ::= **+**  | **-** |
| MulDivOP | ::= **\***  | **/** |
| ComparisonOperator | ::= **<**  | **>**  | **==** |
| Comparison | ::= MathExpr ComparisonOperator MathExpr |
| ConsoleOut | ::= **>>** MathExpr**;** |
| ConsoleIn | ::= **<<** MathExpr**;** |