|  |  |  |
| --- | --- | --- |
| ‹*program*› | →→ | ‹*def-list*› |
| ‹*def-list*› | →→ | *‹def-list-P›* |
| ‹*def-list-P*› | →→ | *<def> ‹def-list-P› | ε* |
| ‹*def*› | →→ | ‹*var-def*› |
| ‹*def*› | →→ | ‹*fun-def*› |
| ‹*var-def*› | →→ | **var** ‹*var-list*› **;** |
| ‹*var-list*› | →→ | ‹*id-list*› |
| ‹*id-list*› | →→ | ‹*id*› ‹*id-list-cont*› |
| ‹*id-list-cont*› | →→ | **,** ‹*id*› ‹*id-list-cont*› |
| ‹*id-list-cont*› | →→ | ε |
| ‹*fun-def*› | →→ | ‹*id*› **(** ‹*param-list*› **)** **{** ‹*var-def-list*› ‹*stmt-list*› **}** |
| ‹*param-list*› | →→ | ‹*id-list*› |
| ‹*param-list*› | →→ | ε |
| ‹*var-def-list*› | →→ | *‹var-def-list-P›* |
| ‹*var-def-list-P*› | →→ | *<var-def> ‹var-def-list-P› | ε* |
| ‹*stmt-list*› | →→ | *‹stmt-list-P›* |
| ‹*stmt-list-P*› | →→ | *<stmt> ‹stmt-list-P› | ε* |
| ‹*stmt*› | →→ | ‹*stmt-assign*› |
| ‹*stmt*› | →→ | ‹*stmt-incr*› |
| ‹*stmt*› | →→ | ‹*stmt-decr*› |
| ‹*stmt*› | →→ | ‹*stmt-fun-call*› |
| ‹*stmt*› | →→ | ‹*stmt-if*› |
| ‹*stmt*› | →→ | ‹*stmt-while*› |
| ‹*stmt*› | →→ | ‹*stmt-do-while*› |
| ‹*stmt*› | →→ | ‹*stmt-break*› |
| ‹*stmt*› | →→ | ‹*stmt-return*› |
| ‹*stmt*› | →→ | ‹*stmt-empty*› |
| ‹*stmt-assign*› | →→ | ‹*id*› **=** ‹*expr*› **;** |
| ‹*stmt-incr*› | →→ | **inc** ‹*id*› **;** |
| ‹*stmt-decr*› | →→ | **dec** ‹*id*› **;** |
| ‹*stmt-fun-call*› | →→ | ‹*fun-call*› **;** |
| ‹*fun-call*› | →→ | ‹*id*› **(** ‹*expr-list*› **)** |
| ‹*expr-list*› | →→ | ‹*expr*› ‹*expr-list-cont*› |
| ‹*expr-list*› | →→ | ε |
| ‹*expr-list-cont*› | →→ | **,** ‹*expr*› ‹*expr-list-cont*› |
| ‹*expr-list-cont*› | →→ | ε |
| ‹*stmt-if*› | →→ | **if** **(** ‹*expr*› **)** **{** ‹*stmt-list*› **}** ‹*else-if-list*› ‹*else*› |
| ‹*else-if-list*› | →→ | *‹else-if-list-P›* |
| ‹*else-if-list-P*› | →→ | **elif** **(** ‹expr› **)** **{** ‹stmt-list› **}** ‹*else-if-list-P*› *| ε* |
| ‹*else*› | →→ | **else** **{** ‹*stmt-list*› **}** |
| ‹*else*› | →→ | ε |
| ‹*stmt-while*› | →→ | **while** **(** ‹*expr*› **)** **{** ‹*stmt-list*› **}** |
| ‹*stmt-do-while*› | →→ | **do** **{** ‹*stmt-list*› **}** **while** **(** ‹*expr*› **)** **;** |
| ‹*stmt-break*› | →→ | **break** **;** |
| ‹*stmt-return*› | →→ | **return** ‹*expr*› **;** |
| ‹*stmt-empty*› | →→ | **;** |
| ‹*expr*› | →→ | ‹*expr-or*› |
| ‹*expr-or*› | →→ | *‹expr-and› ‹expr-or-P›* |
| ‹*expr-or-P*› | →→ | **or** ‹expr-and› ‹expr-or-P› | ε |
| ‹*expr-and*› | →→ | *‹expr-comp›* *‹expr-and-P›* |
| ‹*expr-and-P*› | →→ | **and** *‹expr-comp› ‹expr-and-P› | ε* |
| ‹*expr-comp*› | →→ | *‹expr-rel› ‹expr-comp-P›* |
| ‹*expr-comp-P*› | →→ | *‹op-comp› ‹expr-rel› ‹expr-comp-P› | ε* |
| ‹*op-comp*› | →→ | **==** |
| ‹*op-comp*› | →→ | **<>** |
| ‹*expr-rel*› | →→ | *‹expr-add› ‹expr-rel-P›* |
| ‹*expr-rel-P*› | →→ | *‹op-rel› ‹expr-add› ‹expr-rel-P› | ε* |
| ‹*op-rel*› | →→ | **<** |
| ‹*op-rel*› | →→ | **<=** |
| ‹*op-rel*› | →→ | **>** |
| ‹*op-rel*› | →→ | **>=** |
| ‹*expr-add*› | →→ | *‹expr-mul› ‹expr-add-P›* |
| ‹*expr-add-P*› | →→ | *‹op-add› ‹expr-mul›‹expr-add-P› | ε* |
| ‹*op-add*› | →→ | **+** |
| ‹*op-add*› | →→ | **−** |
| ‹*expr-mul*› | →→ | *‹expr-unary› ‹expr-mul-P›* |
| ‹*expr-mul-P*› | →→ | *‹op-mul› ‹expr-unary› ‹expr-mul-P› | ε* |
| ‹*op-mul*› | →→ | \* |
| ‹*op-mul*› | →→ | **/** |
| ‹*op-mul*› | →→ | **%** |
| ‹*expr-unary*› | →→ | ‹*op-unary*› ‹*expr-unary*› |
| ‹*expr-unary*› | →→ | ‹*expr-primary*› |
| ‹*op-unary*› | →→ | **+** |
| ‹*op-unary*› | →→ | **−** |
| ‹*op-unary*› | →→ | **not** |
| ‹*expr-primary*› | →→ | ‹*id*› |
| ‹*expr-primary*› | →→ | ‹*fun-call*› |
| ‹*expr-primary*› | →→ | ‹*array*› |
| ‹*expr-primary*› | →→ | ‹*lit*› |
| ‹*expr-primary*› | →→ | **(** ‹*expr*› **)** |
| ‹*array*› | →→ | **[** ‹*expr-list*› **]** |
| ‹*lit*› | →→ | ‹*lit-bool*› |
| ‹*lit*› | →→ | ‹*lit-int*› |
| ‹*lit*› | →→ | ‹*lit-char*› |
| ‹*lit*› | →→ | ‹*lit-str*› |