|  |  |  |
| --- | --- | --- |
| **Context Free Grammar for SCOOP** | | |
| Prd # | Productions |  |
|  |  |  |
| 1 | <program> 🡪 <cls\_list> $$ |  |
| 2 | <cls\_list> 🡪 <cls\_list> <class> |  |
| 3 | <cls\_list> 🡪 Є |  |
| 4 | <class> 🡪 <st\_dec\_list> class cls\_id { <cls\_body> } | S |
| 5 | <st\_dec\_list> 🡪 [ state ( <st\_list> ) ] | S |
| 6 | <st\_dec\_list> 🡪 Є | S |
| 7 | <st\_list> 🡪 <st\_list> , <st\_id> | S |
| 8 | <st\_list> 🡪 <st\_id> | S |
| 9 | <st\_id> 🡪 st\_id | S |
| 10 | <cls\_body> 🡪 <var\_dec\_list> < st\_binding> <cls\_m\_list> <c\_s\_m\_list> | S |
| 11 | <var\_dec\_list>🡪<var\_dec\_list> <var\_dec> ; |  |
| 12 | <var\_dec\_list> 🡪 Є |  |
| 13 | <var\_dec>🡪<vd\_start><vd\_rest> |  |
| 14 | <vd\_start> 🡪 <pub-prv><dt> id |  |
| 15 | <dt> 🡪 int |  |
| 16 | <dt> 🡪 string |  |
| 17 | <dt> 🡪 cl\_id |  |
| 18 | <st\_binding> 🡪 statebinding { <b\_list> } | S |
| 19 | <b\_list> 🡪 <b\_list> , <bind> | S |
| 20 | <b\_list> 🡪 <bind> | S |
| 21 | <bind> 🡪 st\_id : ( <rel\_exp> ) | S |
| 22 | <bind> 🡪 st\_id : ( <rel\_exp> ) | S |
| 23 | < st\_binding> 🡪 Є | S |
| 24 | <cls\_m\_list> 🡪 <cls\_m\_list> <cls\_method> |  |
| 25 | <cls\_m\_list> 🡪 Є |  |
| 26 | <cls\_method> 🡪 <pub-prv> <r\_type> <m\_id> ( <arg\_list> ) { <m\_body> } |  |
| 27 | <pub-prv> 🡪 public |  |
| 28 | <pub-prv>🡪 private |  |
| 29 | <r\_type> 🡪 <dt> |  |
| 30 | <m\_id> 🡪 m\_id |  |
| 31 | <arg\_list> 🡪 Є |  |
| 32 | <arg\_list> 🡪 <args> |  |
| 33 | <args>🡪 <args>, <arg> |  |
| 34 | <args>🡪<arg> |  |
| 35 | <arg> 🡪<dt> id |  |
| 36 | <m\_body> 🡪 <var\_dec\_list> <stm\_list> |  |
| 37 | <stm\_list> 🡪 <stm\_list> <stm> |  |
| 38 | <stm\_list> 🡪 Є |  |
| 39 | <stm> 🡪 <lhs> = <exp>; |  |
| 40 | <stm> 🡪 read ( id) ; |  |
| 41 | <stm> 🡪 print <exp>; |  |
| 42 | <stm> 🡪 o\_id.m\_id(<param\_list>); |  |
| 43 | <vd\_new> 🡪 = new ID () |  |
| 44 | <stm> 🡪 if (<rel\_exp>) {<stm\_list> } <else> |  |
| 45 | <else>🡪 else {<stm\_list>} |  |
| 46 | <else> 🡪 Є |  |
| 47 | <stm> 🡪while(<rel\_exp>){<stm\_list> } |  |
| 48 | <stm>🡪for(<init><mid>;<last>) { <stm\_list>} |  |
| 49 | <rel\_exp>🡪<rel\_exp> <rel\_op> <rel\_term> |  |
| 50 | <rel\_term> 🡪<rel\_bool\_fac> |  |
| 51 | <rel\_exp> 🡪<rel\_term> |  |
| 52 | <rel\_term> 🡪 <rel\_fac><comp\_op><rel\_fac> |  |
| 53 | <rel\_bool\_fac> 🡪 bool\_id |  |
| 54 | <rel\_fac> 🡪 id |  |
| 55 | <rel\_fac> 🡪 num |  |
| 56 | < rel\_fac > 🡪 o\_id.p\_id |  |
| 57 | <rel\_op> 🡪 AND |  |
| 58 | <rel\_op> 🡪 OR |  |
| 59 | <comp\_op> 🡪 == |  |
| 60 | <comp\_op> 🡪 != |  |
| 61 | <comp\_op> 🡪 > |  |
| 62 | <comp\_op> 🡪 < |  |
| 63 | <comp\_op> 🡪 <= |  |
| 64 | <comp\_op> 🡪 >= |  |
| 65 | <exp> 🡪 <term> |  |
| 66 | <exp> 🡪 <exp> <add\_op> <term> |  |
| 67 | <term> 🡪 <factor> |  |
| 68 | <term> 🡪 <term> <mult\_op> <factor> |  |
| 69 | <factor> 🡪 ( <exp> ) |  |
| 70 | <factor> 🡪 id |  |
| 71 | <factor> 🡪 o\_id.p\_id |  |
| 72 | <factor> 🡪 num |  |
| 73 | <add\_op> 🡪 + |  |
| 74 | <add\_op> 🡪 - |  |
| 75 | <mult\_op> 🡪\* |  |
| 76 | <mult\_op> 🡪/ |  |
| 77 | <c\_s\_m\_list>🡪 Є |  |
| 78 | <c\_s\_m\_list>🡪<c\_s\_m\_list> <c\_s\_method> | S |
| 79 | <c\_s\_method> 🡪 <s\_list> { <s\_v\_d\_lst><s\_m\_list> } | S |
| 80 | <s\_m\_list> 🡪<s\_m\_list> <s\_method> | S |
| 81 | <s\_m\_list> 🡪 <s\_method> | S |
| 82 | <s\_method>🡪<r\_type><m\_id>(<arg\_list>)<s\_end\_list>{<s\_m\_body>} | S |
| 83 | <s\_end\_list> 🡪 [<s\_e\_list>] | S |
| 84 | <s\_list> 🡪 <s\_list> , <s\_id> | S |
| 85 | <s\_list> 🡪 <s\_id> | S |
| 86 | <s\_e\_list> 🡪<s\_e\_list> | st\_id | S |
| 87 | <s\_e\_list> 🡪 st\_id | S |
| 88 | <s\_m\_body> 🡪<var\_dec\_list> <s\_stm\_list> |  |
| 89 | <s\_stm\_list> 🡪<s\_stm\_list> <st\_stm> |  |
| 90 | <s\_stm\_list> 🡪 Є |  |
| 91 | <st\_stm>🡪this.trans(st\_id); | S |
| 92 | <st\_stm>🡪<lhs> = <exp>; |  |
| 93 | <st\_stm>🡪 read ( id) ; |  |
| 94 | <st\_stm>🡪 print <exp>; |  |
| 95 | <st\_stm> 🡪o\_id.m\_id(<param\_list>); |  |
| 96 | <st\_stm> 🡪o\_id.m\_id(<param\_list>); |  |
| 97 | <st\_stm>🡪 while(<rel\_exp>){<s\_stm\_list> } |  |
| 98 | <st\_stm>🡪 for(<init><mid>;<last>) { <s\_stm\_list> } |  |
| 99 | < st\_stm> 🡪 if (<rel\_exp>) {<s\_stm\_list>}<s\_else> |  |
| 100 | <s\_else> 🡪 else {<s\_stm\_list>} |  |
| 101 | <s\_else> 🡪 Є |  |
| 102 | <init>🡪 id = num; |  |
| 103 | <mid>🡪<rel\_exp> |  |
| 104 | <last>🡪id <in\_dec> |  |
| 105 | <in\_dec>🡪 ++ |  |
| 106 | <in\_dec>🡪 -- |  |
| 107 | <lhs>🡪id |  |
| 108 | <lhs>🡪 o\_id.p\_id |  |
| 109 | <param\_list>🡪<params> |  |
| 110 | <params>🡪<params>,<factor> |  |
| 111 | <params>🡪<factor> |  |
| 112 | <param\_list>🡪 Є |  |
| 113 | <new>🡪 Є |  |
| 114 | <cl\_id>🡪 cl\_id |  |
| 115 | <s\_id> 🡪 [st\_id] | S |
| 116 | <s\_list> 🡪 [st\_id]:[ st\_id] | S |
| 117 | <vd\_rest> 🡪 <vd\_simple> | S |
| 118 | <vd\_rest> 🡪<vd\_new> | S |
| 119 | <vd\_simple> 🡪 <vd\_simple> , id | S |
| 120 | <vd\_simple> 🡪 Є | S |
| 121 | <vd\_new> 🡪 Є | S |
| 122 | <s\_v\_d\_lst> 🡪 <s\_v\_d\_lst><s\_v\_d> ; | S |
| 123 | <s\_v\_d\_lst> 🡪 Є | S |
| 124 | <s\_v\_d>🡪<s\_vd\_start><vd\_rest> | S |
| 125 | < s\_vd\_start> 🡪 <dt> id | S |
|  |  |  |
|  |  |  |