1. <START> 🡪<DEFS>
2. <DEFS> 🡪 <MST> <DEFS> | <CLASS> <DEFS> | <FUNCTION-DEC><DEFS>
3. <MST> 🡪 € |<SST> <MST>
4. <SST ***>*** 🡪 <SST1>
   1. <SST1 >🡪 <DEC> | <WHILE> | <FOR > | <DO-WHILE> | <CONST-DT> | <INC-DEC-PRE> | <IF-ELSE> |<SWITCH-CASE> | <RETURN> |<GTSWID>
5. <WHILE> 🡪 while (<EXP>) <BODY>
6. <FOR> 🡪 for (<FOR-PARAM>) <BODY>
   1. <FOR-PARAM> 🡪 ID <FOR-PARAM-2> | <DEC>;<C2>;<C3> |; <C2>;<C3>
   2. <FOR-PARAM-2> 🡪 ; <C2>;<C3> | <INIT-VALUE>;<C2>;<C3> | <RANGE-FOR>
   3. <RANGE-FOR> 🡪 in <FOR\_R>
   4. <FOR\_R> 🡪 ID <N\_INIT\_VALUE>
   5. <C2> 🡪 <EXP> | €
   6. <C3> 🡪 ID | <GTSWID>
7. <DO-WHILE> 🡪 do <BODY> while (<EXP>)
8. <BODY> 🡪 ; | <SST> | {<MST>}
9. <IF-ELSE> 🡪 if (<COND>) <BODY> <ELSE>
   1. <ELSE>🡪 € | else <BODY>
10. <CONST-DT> 🡪 const ID <CONST-DT1> <NEXT-CONST-DT>
    1. <CONST-DT1> 🡪 AOR <CONST-DT2>
    2. <CONST-DT2> 🡪 new ID ( <CALLING-PARAM> ) | <INIT-VALUE-2> | ID <INIT-VALUE>
    3. <NEXT-CONST-DT> 🡪 <DEC\_PARAMS> | €
11. <GTSWID> 🡪 ID <INIT\_VALUE> <NEXT\_GTSWID>
    1. <NEXT\_GTSWID> 🡪 inc\_dec | ( <CALLING\_PARAM> ) | AOR <GTSDEC>
    2. <GTSDEC> 🡪 <DEC2> | <OBJECT> | <ARRAY>
12. <FUNCTION-DEC>🡪 function <FUNC-DEF-1>
    1. <FUNC-DEF-1>🡪ID (<DEC-PARAMS>) {<MST>}
13. <CLASS> 🡪 class ID <CLASS-STRUCT>
    1. <CLASS-STRUCT> 🡪 <CLASS-BODY> | : ID <CLASSBODY>
    2. <CLASSBODY>🡪 { <CLASS-MST> }
    3. <CLASS-MST> 🡪 € |

<CLASS\_CONSTRUCOR> <CLASS-ST> <CLASS-MST>

* + 1. <CLASS\_CONSTRUCOR> 🡪 € | constructor (<DEC\_PARAMS>) {<CONSTRUCTOR\_BODY>}
    2. <CONSTRUCTOR\_BODY> 🡪 € | <C\_B\_I> <CONSTRUCTOR\_BODY>
    3. <C\_B\_I> 🡪 <THIS\_VAR>|<SST1>
    4. <THIS\_VAR > 🡪 this. ID AOR <DEC2>
  1. <CLASS-ST> 🡪 <CLASS-FUNC> | <DEC> | <PROTECTED>
  2. <CLASS-FUNC> 🡪 ID <FUNC-DEF>
  3. <FUNC-DEF> 🡪 AOR ( <DEC-PARAMS> ) => { <MST> } |

(<DEC-PARAMS>){<MST>}

* 1. <PROTECTED>🡪 protected: <PRO-NEXT>
  2. <PRO-NEXT> 🡪 <CLASS-ST> | { <PRO-BODY>}
  3. <PRO-BODY> 🡪 <CLASS-MST>

1. <DEC\_PARAMS > 🡪 DT ID <DEC1><NEXT\_PARAM> |

ID <CONST\_DEC\_PARAM> <NEXT\_PARAM> |

€

* 1. <CONST\_DEC\_PARAM> 🡪 € | AOR <CONST\_DT1>
  2. <NEXT\_PARAM> 🡪 , <NEXT\_DEC\_PARAM> | €
  3. <NEXT\_DEC\_PARAM> 🡪 DT ID <DEC1><NEXT\_PARAM> |

ID <CONST\_DEC\_PARAM> <NEXT\_PARAM>

1. <Dec> 🡪 DT ID <DEC1> <NEXTDEC>
   1. <DEC1> 🡪 AOR <DEC2> | €
   2. <DEC2> 🡪 ID <DEC3> | <E><EXP1>
   3. <DEC3> 🡪 <MERGED\_INIT> <EXP1> | <EXP1> | €
   4. <MERGED\_INIT> 🡪 <MOV> | [<EXP>] <M\_N\_ARR>
   5. <M\_N\_ARR> 🡪 <MOV> | [<EXP>] <M\_N\_ARR2> | €
   6. <M\_N\_ARR2> 🡪 <MOV> | €
   7. <MOV> 🡪 .ID<MERGED1> | (<CALLING\_PARAMS>) <MERGED>
   8. <MERGED1> 🡪 <EXP1> | <MERGED\_INIT> <DEC1>
   9. <MERGED> 🡪 <EXP1> | <MERGED\_INIT>
2. <EXP> 🡪 <VAL> <EXP1>
   1. <EXP1> 🡪 <Q\_DASH> <R\_DASH> <S\_DASH><T\_DASH> <E\_DASH>
   2. <E\_DASH> 🡪 ||<T><E\_DASH> | €
   3. <T> 🡪 <S> <T\_DASH>
   4. <T\_DASH> 🡪 && <S><T\_DASH> | €
   5. <S> 🡪 <R> <S\_DASH>
   6. <S\_DASH> 🡪 ROP <R> <S\_DASH> | €
   7. <R> 🡪 <Q> <R\_DASH>
   8. <R\_DASH> 🡪 PM <Q> <R\_DASH> | €
   9. <Q> 🡪 <VAL> <Q\_DASH>
   10. <Q\_DASH>🡪 MDM <VAL> <Q\_DASH> | €
   11. <VAL> 🡪 <F> | <E>
   12. <F> 🡪 ID <NEW\_ASGN>
   13. <E> 🡪 (<EXP >) |! <VAL > | -- <F > | ++ <F > | <CONST > |this . ID <NEW\_ASGN>
   14. <NEW\_ASGN > 🡪 ++ | -- | <N\_INIT\_VALUE>
3. <N\_INIT\_VALUE>🡪€ | <OTHER\_N\_VALUE> | [<EXP>] <N\_ARR\_N>
   1. <N\_ARR\_N>🡪 € | <OTHER\_N\_VALUE> | [<EXP>] <N\_ARR\_2N>
   2. <N\_ARR\_2N> 🡪 € | <OTHER\_N\_VALUE>
   3. <OTHER\_N\_VALUE> 🡪 .ID <N\_INIT\_VALUE> | (<CALLING\_PARAM>) <N\_INIT\_VALUE>
4. <INIT\_VALUE> 🡪 € | <OTHER\_VALUE> | [ <EXP> ] <N\_ARR>
   1. <OTHER\_VALUE> 🡪 . ID <INIT\_VALUE> | ( <CALLING\_PARAM>) <AFT\_VALUE>
   2. <AFT\_VALUE> 🡪 <OTHER\_VALUE> | [ <EXP> ] <N\_ARR>
   3. <N\_ARR> 🡪 € | <OTHER\_VALUE> | [ <EXP> ] <N\_ARR2>
   4. <N\_ARR2> 🡪 € | <OTHER\_VALUE>
5. <ARRAY> 🡪 [ <ARRAY-INNER>]
   1. <ARRAY-INNER> 🡪 € | <ARRAY-VALUES>
   2. <ARRAY-VALUES> 🡪<ARRAY-VAL><NEXT-VAL>
   3. <ARRAY-VAL> 🡪 <INIT-VALUE-2> | … ID <N\_INIT-VALUE>
   4. <NEXT-VAL>🡪 , <ARRAY-VALUES> | €
6. <SPREAD> 🡪 … ID <N-INIT-VALUE>
7. <OBJECT> 🡪 {<PROP>}
   1. <PROP>🡪 <OBJECT1> <NEXT-PROP>
   2. <OBJECT1> 🡪 ID : <EXP> | <SPREAD>
   3. <NEXT-PROP>🡪, <PROP>| €

1. <INIT-VALUE-2> 🡪 <EXP> | <ARRAY> | <OBJECT>
2. <CALLING-PARAMS> 🡪 <CP-VALUE>
   1. <CP-VALUE> 🡪 € | <CP-VALUE2>
   2. <CP-VALUE2> 🡪 <INIT-VALUE-2> <NEXT-CPVALUE>
   3. <NEXT-CPVALUE> 🡪 € |, <CP-VALUE2>
3. <SWITCH-CASE> 🡪 switch (<N-INIT-VALUE>) <SW-BODY>
   1. <SW-BODY> 🡪 {<CASES> <DEFAULT>}
   2. <CASES>🡪 case <CASE-VALUE>: <BODY>
   3. <CASE-VALUE> 🡪 <INNER-CASE-VAL>
   4. <INNER-CASE-VAL> 🡪 <EXP>
   5. <DEFAULT> 🡪 default : <BODY>
4. <INC-DEC-PRE> 🡪 <inc-dec-op> ID <N-INIT-VALUE>