LEGACY ASM286 GRAMMAR

All whiteSpace before a terminal is ignored by the assembler. The assembler is case-insensitive for all except identifiers (id). The ***empty*** terminal means no text.

|  |  |  |
| --- | --- | --- |
| **Non-Terminal** | **Definition** | **Notes** |
| newline | linefeed |  |
|  | | comment |  |
|  |  |  |
| linefeed | ASCII 10 |  |
|  |  |  |
| comment | **;** text linefeed |  |
|  |  |  |
| whiteSpaceCharacter | ASCII 8, 9, 11-13, 26, 32 |  |
|  | | **\** text linefeed lineDir |  |
|  | | **\** text linefeed |  |
|  |  |  |
| whiteSpace | whitespace whiteSpaceCharacter |  |
|  | | whiteSpaceCharacter |  |
|  |  |  |
| text | text character |  |
|  | | character |  |
|  |  |  |
| character | Any character in the ordinal range 0-255 except linefeed |  |
|  |  |  |
| module | directiveList endDir |  |
|  |  |  |
| endDir | **END** immExpr newline |  |
|  | | **END** newline |  |
|  |  |  |
| directiveList | directiveList directive |  |
|  | | directive |  |
|  | | ***empty*** |  |
|  |  |  |
| directive | generalDir |  |
|  | | segmentDef |  |
|  | | newline |  |
|  |  |  |
| segmentDef | segmentDir inSegDirList endsDir |  |
|  | | simpleSegDir inSegDirList simpleEndsDir |  |
|  |  |  |
| simpleSegDir | segDir newline |  |
|  |  |  |
| segDir | **.CODE** segId |  |
|  | | **.CODE** |  |
|  | | **.DATA?** |  |
|  | | **.DATA** |  |
|  | | **.CONST** |  |
|  | | **.FARDATA?** segId |  |
|  | | **.FARDATA?** |  |
|  | | **.FARDATA** segId |  |
|  | | **.FARDATA** |  |
|  | | **.STACK** constExpr |  |
|  | | **.STACK** |  |
|  |  |  |
| endsDir | segId **ENDS** newline |  |
|  |  |  |
| simpleEndsDir | endsDir |  |
|  | | ***empty*** |  |
|  |  |  |
| segmentDir | segId **SEGMENT** segOptionList newline |  |
|  |  |  |
| segId | id |  |
|  |  |  |
| id | id idAlphaNumeric |  |
|  | | idAlpha |  |
|  |  |  |
| idAlpha | alpha |  |
|  | | underscore character (\_) |  |
|  | | at character (@) |  |
|  | | question mark character (?) |  |
|  |  |  |
| idAlphaNumeric | idAlpha |  |
|  | | decDigit |  |
|  |  |  |
| alpha | any character between ‘A’ and ‘Z’ or ‘a’ and ‘z’ |  |
|  |  |  |
| decDigit | any character between ‘0’ and ‘9’ |  |
|  |  |  |
| segOptionList | segOptionList segOption |  |
|  | | segOption |  |
|  | | ***empty*** |  |
|  |  |  |
| segOption | segAlign |  |
|  | | segAttrib |  |
|  | | className |  |
|  | | **READONLY** |  |
|  |  |  |
| segAlign | **BYTE** |  |
|  | | **WORD** |  |
|  | | **DWORD** |  |
|  | | **PARA** |  |
|  | | **PAGE** |  |
|  |  |  |
| segAttrib | **PUBLIC** |  |
|  | | **STACK** |  |
|  | | **COMMON** |  |
|  | | **MEMORY** |  |
|  | | **AT** constExpr |  |
|  | | **PRIVATE** |  |
|  |  |  |
| className | string |  |
|  |  |  |
| string | quote stext quote |  |
|  |  |  |
| quote | single quote character (‘) |  |
|  |  |  |
| stext | stext stringChar |  |
|  | | stringChar |  |
|  |  |  |
| stringChar | quote quote |  |
|  | | any character except quote |  |
|  |  |  |
| inSegDirList | inSegDirList inSegDir |  |
|  | | inSegDir |  |
|  | | ***empty*** |  |
|  |  |  |
| inSegDir | labelDef inSegmentDir |  |
|  | | labelDef newline |  |
|  | | inSegmentDir |  |
|  |  |  |
| labelDef | id **:** |  |
|  | | id **::** |  |
|  |  |  |
| inSegmentDir | instruction | \*\*\* there are more |
|  | | fpInstruction |  |
|  | | dataDir |  |
|  | | offsetDir |  |
|  | | labelDir |  |
|  | | generalDir |  |
|  | | segmentDef |  |
|  |  |  |
| instruction | instrPrefix asmInstruction |  |
|  | | asmInstruction |  |
|  |  |  |
| dataDir | id dataItem newline |  |
|  | | dataItem newline |  |
|  |  |  |
| dataItem | dataDecl scalarInstList | \*\*\* there are more |
|  |  |  |
| dataDecl | **DB** |  |
|  | | **DW** |  |
|  | | **DD** |  |
|  | | **DF** |  |
|  | | **DQ** |  |
|  | | **DT** |  |
|  | | dataType |  |
|  |  |  |
| scalarInstList | scalarInstList commaOptNewline initValue |  |
|  | | initValue |  |
|  |  |  |
| commaOptNewline | **,** newline |  |
|  | | **,** |  |
|  |  |  |
| initValue | immExpr |  |
|  | | string |  |
|  | | **?** |  |
|  | | constExpr **DUP** **(** scalarInstList **)** |  |
|  | | floatNumber |  |
|  | | bcdConstant |  |
|  |  |  |
| dataType | **BYTE** |  |
|  | | **SBYTE** |  |
|  | | **WORD** |  |
|  | | **SWORD** |  |
|  | | **DWORD** |  |
|  | | **SDWORD** |  |
|  | | **FWORD** |  |
|  | | **QWORD** |  |
|  | | **TBYTE** |  |
|  | | **REAL4** |  |
|  | | **REAL8** |  |
|  | | **REAL10** |  |
|  |  |  |
| floatNumber | [sign] decNumber **.** [decNumber] [exponent] | \*\*\* Also R variant |
|  |  |  |
| sign | **+** |  |
|  | | **-** |  |
|  |  |  |
| decNumber | decNumber decDigit |  |
|  | | decDigit |  |
|  |  |  |
| exponent | **E** sign decNumber |  |
|  | | **E** decNumber |  |
|  |  |  |
| bcdConstant | sign decNumber |  |
|  | | decNumber |  |
|  |  |  |
| instPrefix | **REP** |  |
|  | | **REPE** |  |
|  | | **REPZ** |  |
|  | | **REPNE** |  |
|  | | **REPNZ** |  |
|  | | **LOCK** |  |
|  |  |  |
| offsetDir | offsetDirType newline |  |
|  |  |  |
| offsetDirType | **EVEN** |  |
|  | | **ORG** immExpr |  |
|  | | **ALIGN** constExpr |  |
|  |  |  |
| labelDir | id **LABEL** qualifiedType newline |  |
|  |  |  |
| qualifiedType | nearfar **PTR** type | \*\*\* there are more |
|  | | type |  |
|  |  |  |
| nearfar | **NEAR** |  |
|  | | **FAR** |  |
|  |  |  |
| type | dataType |  |
|  |  |  |
| generalDir | modelDir | \*\*\* there are more |
|  | | nameDir |  |
|  | | groupDir |  |
|  | | assumeDir |  |
|  | | externDir |  |
|  | | publicDir |  |
|  | | equDir |  |
|  | | textEquDir |  |
|  | | equalDir |  |
|  | | ifDir |  |
|  | | optionDir |  |
|  | | processorDir |  |
|  | | titleDir |  |
|  | | pageDir |  |
|  | | listDir |  |
|  | | echoDir |  |
|  | | lineDir |  |
|  |  |  |
| modelDir | **.MODEL** memOption **,** modelOptList newline |  |
|  | | **.MODEL** memOption newline |  |
|  |  |  |
| memOption | **TINY** |  |
|  | | **SMALL** |  |
|  | | **MEDIUM** |  |
|  | | **COMPACT** |  |
|  | | **LARGE** |  |
|  | | **HUGE** | Same as LARGE |
|  |  |  |
| modelOptList | modelOptList **,** modelOpt |  |
|  | | modelOpt |  |
|  |  |  |
| modelOpt | langType |  |
|  | | stackOption |  |
|  |  |  |
| langType | **C** | There are others |
|  | | **PASCAL** |  |
|  |  |  |
| stackOption | **NEARSTACK** |  |
|  | | **FARSTACK** |  |
|  |  |  |
| nameDir | **NAME** id newline |  |
|  |  |  |
| fileSpec | string |  |
|  |  |  |
| groupDir | groupId **GROUP** segIdList |  |
|  |  |  |
| groupId | id |  |
|  |  |  |
| segIdList | segIdList **,** segId |  |
|  | | segId |  |
|  |  |  |
| assumeDir | **ASSUME** assumeList newline |  |
|  | | **ASSUME** **NOTHING** newline |  |
|  |  |  |
| assumeList | assumeList assumeRegister |  |
|  | | assumeRegister |  |
|  |  |  |
| assumeRegister | assumeSegReg |  |
|  | | assumeReg |  |
|  |  |  |
| assumeSegReg | segmentRegister **:** assumeSegVal |  |
|  |  |  |
| assumeSegVal | frameExpr |  |
|  | | **NOTHING** |  |
|  | | **ERROR** |  |
|  |  |  |
| frameExpr | **SEG** id |  |
|  | | **DGROUP** **:** id |  |
|  | | segmentRegister **:** id |  |
|  | id |  |
|  |  |  |
| segmentRegister | **CS** |  |
|  | | **DS** |  |
|  | | **ES** |  |
|  | | **SS** |  |
|  |  |  |
| assumeReg | register **:** assumeVal |  |
|  |  |  |
| register | gpRegister |  |
|  | | byteRegister |  |
|  |  |  |
| gpRegister | **AX** |  |
|  | | **BX** |  |
|  | | **CX** |  |
|  | | **DX** |  |
|  | | **BP** |  |
|  | | **SP** |  |
|  | | **DI** |  |
|  | | **SI** |  |
|  |  |  |
| byteRegister | **AL** |  |
|  | | **AH** |  |
|  | | **BL** |  |
|  | | **BH** |  |
|  | | **CL** |  |
|  | | **CH** |  |
|  | | **DL** |  |
|  | | **DH** |  |
|  |  |  |
| assumeVal | qualifiedType |  |
|  | | **NOTHING** |  |
|  | | **ERROR** |  |
|  |  |  |
| externDir | **EXTERN** externList newline |  |
|  |  |  |
| externList | externList commaOptNewline externDef |  |
|  | | externDef |  |
|  |  |  |
| externDef | langType id **:** externType |  |
|  | | id **:** externType |  |
|  |  |  |
| externType | **ABS** |  |
|  | | qualifiedType |  |
|  |  |  |
| publicDir | **PUBLIC** pubList newline |  |
|  |  |  |
| pubList | pubList commaOptNewline pubDef |  |
|  | | pubDef |  |
|  |  |  |
| pubDef | langType id |  |
|  | | id |  |
|  |  |  |
| equDir | equId **EQU** immExpr newline |  |
|  |  |  |
| equId | id |  |
|  |  |  |
| textEquDir | equId **TEXTEQU** textList newline |  |
|  |  |  |
| textList | textList commaOptNewline textItem |  |
|  | | textItem |  |
|  |  |  |
| textItem | **%** constExpr |  |
|  | | string |  |
|  | | equId |  |
|  |  |  |
| equalDir | id **=** immExpr newline |  |
|  |  |  |
| processor | **.8086** |  |
|  | | **.186** |  |
|  | | **.286** |  |
|  | | **.286P** |  |
|  |  |  |
| processorDir | processor newline |  |
|  | | coprocessor newline |  |
|  |  |  |
| coprocessor | **.8087** |  |
|  | | **.287** |  |
|  | | **.NO87** |  |
|  |  |  |
| titleDir | titleType arbitaryText newline |  |
|  |  |  |
| titleType | **TITLE** |  |
|  | | **SUBTITLE** |  |
|  |  |  |
| arbitaryText | text |  |
|  |  |  |
| pageDir | **PAGE** pageExpr newline |  |
|  | | **PAGE** newline |  |
|  |  |  |
| pageExpr | **+** |  |
|  | | pageLength **,** pageWidth |  |
|  | | pageLength |  |
|  |  |  |
| pageLength | constExpr |  |
|  |  |  |
| pageWidth | constExpr |  |
|  |  |  |
| listDir | listOption newline |  |
|  |  |  |
| listOption | **.LIST** | \*\*\* there are other options |
|  | | **.NOLIST** |  |
|  |  |  |
| echoDir | **ECHO** textList newline |  |
|  | | **ECHO** arbitaryText newline |  |
|  |  |  |
| e01 | e01 orOp e02 |  |
|  | | e02 |  |
|  |  |  |
| orOp | **OR** |  |
|  | | **XOR** |  |
|  |  |  |
| e02 | e02 andOp e03 |  |
|  | | e03 |  |
|  |  |  |
| andOp | **AND** |  |
|  |  |  |
| e03 | **NOT** e04 |  |
|  | | e04 |  |
|  |  |  |
| e04 | e04 relOp e05 |  |
|  | | e05 |  |
|  |  |  |
| relOp | **EQ** |  |
|  | | **NE** |  |
|  | | **LT** |  |
|  | | **LE** |  |
|  | | **GT** |  |
|  | | **GE** |  |
|  |  |  |
| e05 | e05 addOp e06 |  |
|  | | e06 |  |
|  |  |  |
| addOp | **+** |  |
|  | | **-** |  |
|  |  |  |
| e06 | e06 mulOp e07 |  |
|  | | e07 |  |
|  |  |  |
| mulOp | **\*** |  |
|  | | **/** |  |
|  | | **MOD** |  |
|  | | **SHR** |  |
|  | | **SHL** |  |
|  |  |  |
| e07 | addOp e08 |  |
|  | | e08 |  |
|  |  |  |
| e08 | e08Op e09 |  |
|  | | e09 |  |
|  |  |  |
| e08Op | **HIGH** |  |
|  | | **LOW** |  |
|  |  |  |
| e09 | **OFFSET** e10 | \*\*\* there are more |
|  | | **SEG** e10 |  |
|  | | **TYPE** e10 |  |
|  | | **THIS** e10 |  |
|  | | e10 |  |
|  |  |  |
| e10 | e10 **[** expr **]** |  |
|  | | e11 |  |
|  |  |  |
| e11 | | **SIZEOF** sizeArg | \*\*\* there are more |
|  | | **LENGTHOF** id |  |
|  | | constant |  |
|  | | string |  |
|  | | **DEFINED** id |  |
|  | | **$** |  |
|  | | segmentRegister |  |
|  | | register |  |
|  | | **ST** **(** expr **)** |  |
|  | | **ST** |  |
|  | |id |  |
|  | **|** e12 |  |
| e12 | **(** expr **)** |  |
|  | | **[** expr **]** |  |
|  |  |  |
| expr | **SHORT** e01 |  |
|  | | e01 |  |
|  |  |  |
| constExpr | expr |  |
|  |  |  |
| immExpr | expr |  |
|  |  |  |
| sizeArg | type | \*\*\* there are more |
|  | | id |  |
|  |  |  |
| constant | sign **decNumber** |  |
|  | | **decNumber** |  |
|  | | **hexNumber** |  |
|  | | **octNumber** |  |
|  | | **binNumber** |  |
|  | | bool |  |
|  |  |  |
| bool | **FALSE** |  |
|  | | **TRUE** |  |
|  |  |  |
| asmInstruction | mnemonicTwo operand **,** operand newline |  |
|  | | mnemonicOne operand newline |  |
|  | | mnemonicZero newline |  |
|  |  |  |
| fpInstruction | fpMnemonicTwo operand **,** operand newline |  |
|  | | fpMnemonicOne operand newline |  |
|  | | fpMnemonicZero newline |  |
|  |  |  |
| operand | expr |  |
|  |  |  |
| mnemonicZero | **AAA** |  |
|  | | **AAD** |  |
|  | | **AAM** |  |
|  | | **AAS** |  |
|  | | **CBW** |  |
|  | | **CLC** |  |
|  | | **CLTS** |  |
|  | | **CLD** |  |
|  | | **CLI** |  |
|  | | **CMC** |  |
|  | | **CWD** |  |
|  | | **DAA** |  |
|  | | **DAS** |  |
|  | | **HLT** |  |
|  | | **INTO** |  |
|  | | **IRET** |  |
|  | | **LAHF** |  |
|  | | **LEAVE** |  |
|  | | **NOP** |  |
|  | | **POPA** |  |
|  | | **POPF** |  |
|  | | **PUSHA** |  |
|  | | **PUSHF** |  |
|  | | **RET** |  |
|  | | **SAHF** |  |
|  | | **STC** |  |
|  | | **STD** |  |
|  | | **STI** |  |
|  | | **WAIT** |  |
|  |  |  |
| mnemonicOne | **CALL** |  |
|  | | **DEC** |  |
|  | | **DIV** |  |
|  | | **IDIV** |  |
|  | | **IMUL** |  |
|  | | **INC** |  |
|  | | **INT** |  |
|  | | **JA** |  |
|  | | **JNBE** |  |
|  | | **JAE** |  |
|  | | **JNB** |  |
|  | | **JB** |  |
|  | | **JNAE** |  |
|  | | **JBE** |  |
|  | | **JNA** |  |
|  | | **JC** |  |
|  | | **JCXZ** |  |
|  | | **JE** |  |
|  | | **JZ** |  |
|  | | **JG** |  |
|  | | **JNLE** |  |
|  | | **JGE** |  |
|  | | **JNL** |  |
|  | | **JL** |  |
|  | | **JNGE** |  |
|  | | **JLE** |  |
|  | | **JNG** |  |
|  | | **JMP** |  |
|  | | **JNC** |  |
|  | | **JNE** |  |
|  | | **JNZ** |  |
|  | | **JNO** |  |
|  | | **JNS** |  |
|  | | **JNP** |  |
|  | | **JPO** |  |
|  | | **JO** |  |
|  | | **JP** |  |
|  | | **JPE** |  |
|  | | **JS** |  |
|  | | **LGDT** |  |
|  | | **LIDT** |  |
|  | | **LLDT** |  |
|  | | **LMSW** |  |
|  | | **LODSB** |  |
|  | | **LODSW** |  |
|  | | **LODS** |  |
|  | | **LOOP** |  |
|  | | **LOOPE** |  |
|  | | **LOOPZ** |  |
|  | | **LOOPNZ** |  |
|  | | **LOOPNE** |  |
|  | | **LTR** |  |
|  | | **MOVSB** |  |
|  | | **MOVSW** |  |
|  | | **MOVS** |  |
|  | | **MUL** |  |
|  | | **NEG** |  |
|  | | **NOT** |  |
|  | | **POP** |  |
|  | | **PUSH** |  |
|  | | **RET** |  |
|  | | **SCASB** |  |
|  | | **SCASW** |  |
|  | | **SCAS** |  |
|  | | **SGDT** |  |
|  | | **SIDT** |  |
|  | | **SLDT** |  |
|  | | **SMSW** |  |
|  | | **STOSB** |  |
|  | | **STOSW** |  |
|  | | **STOS** |  |
|  | | **STR** |  |
|  | | **VERR** |  |
|  | | **VERW** |  |
|  | | **XLAT** |  |
|  | | **XLATB** |  |
|  |  |  |
| mnemonicTwo | **ADC** |  |
|  | | **ADD** |  |
|  | | **AND** |  |
|  | | **ARPL** |  |
|  | | **BOUND** |  |
|  | | **CMP** |  |
|  | | **CMPSW** |  |
|  | | **CMPSB** |  |
|  | | **CMPS** |  |
|  | | **ENTER** |  |
|  | | **ESC** |  |
|  | | **IN** |  |
|  | | **INSB** |  |
|  | | **INSW** |  |
|  | | **INS** |  |
|  | | **LAR** |  |
|  | | **LDS** |  |
|  | | **LEA** |  |
|  | | **LES** |  |
|  | | **LSL** |  |
|  | | **MOV** |  |
|  | | **OR** |  |
|  | | **OUT** |  |
|  | | **OUTSB** |  |
|  | | **OUTSW** |  |
|  | | **OUTS** |  |
|  | | **RCL** |  |
|  | | **RCR** |  |
|  | | **ROL** |  |
|  | | **ROR** |  |
|  | | **SAL** |  |
|  | | **SHL** |  |
|  | | **SAR** |  |
|  | | **SBB** |  |
|  | | **SHR** |  |
|  | | **SUB** |  |
|  | | **TEST** |  |
|  | | **XCHG** |  |
|  | | **XOR** |  |
|  |  |  |
| fpMnemonicTwo | **FADD** |  |
|  | | **FADDP** |  |
|  | | **FDIV** |  |
|  | | **FDIVP** |  |
|  | | **FDIVR** |  |
|  | | **FDIVRP** |  |
|  | | **FMUL** |  |
|  | | **FMULP** |  |
|  | | **FSUB** |  |
|  | | **FSUBP** |  |
|  | | **FSUBR** |  |
|  | | **FSUBRP** |  |
|  |  |  |
| fpMnemonicOne | **FADD** |  |
|  | | **FBLD** |  |
|  | | **FBSTP** |  |
|  | | **FCOM** |  |
|  | | **FCOMP** |  |
|  | | **FDIV** |  |
|  | | **FDIVR** |  |
|  | | **FFREE** |  |
|  | | **FIADD** |  |
|  | | **FICOM** |  |
|  | | **FICOMP** |  |
|  | | **FIDIV** |  |
|  | | **FIDIVR** |  |
|  | | **FILD** |  |
|  | | **FIMUL** |  |
|  | | **FIST** |  |
|  | | **FISTP** |  |
|  | | **FISUB** |  |
|  | | **FISUBR** |  |
|  | | **FLD** |  |
|  | | **FLDCW** |  |
|  | | **FLDENV** |  |
|  | | **FMUL** |  |
|  | | **FRSTOR** |  |
|  | | **FSAVE** |  |
|  | | **FNSAVE** |  |
|  | | **FST** |  |
|  | | **FSTCW** |  |
|  | | **FNSTCW** |  |
|  | | **FSTENV** |  |
|  | | **FNSTENV** |  |
|  | | **FSTP** |  |
|  | | **FSTSW** |  |
|  | | **FNSTSW** |  |
|  | | **FSUB** |  |
|  | | **FSUBR** |  |
|  | | **FXCH** |  |
|  |  |  |
| fpMnemonicZero | **FABS** |  |
|  | | **FCHS** |  |
|  | | **FCLEX** |  |
|  | | **FNCLEX** |  |
|  | | **FCOMPP** |  |
|  | | **FDECSTP** |  |
|  | | **FDISI** |  |
|  | | **FNDISI** |  |
|  | | **FENI** |  |
|  | | **FNENI** |  |
|  | | **FINCSTP** |  |
|  | | **FINIT** |  |
|  | | **FNINIT** |  |
|  | | **FLDLG2** |  |
|  | | **FLDLN2** |  |
|  | | **FLDL2E** |  |
|  | | **FLDL2T** |  |
|  | | **FLDPI** |  |
|  | | **FLDZ** |  |
|  | | **FLD1** |  |
|  | | **FNOP** |  |
|  | | **FPATAN** |  |
|  | | **FPREM** |  |
|  | | **FPTAN** |  |
|  | | **FRNDINT** |  |
|  | | **FSCALE** |  |
|  | | **FSETPM** |  |
|  | | **FSQRT** |  |
|  | | **FTST** |  |
|  | | **FWAIT** |  |
|  | | **FXAM** |  |
|  | | **FXTRACT** |  |
|  | | **FYL2X** |  |
|  | | **FYL2XP1** |  |
|  | | **F2XM1** |  |
|  |  |  |
| optionDir | **OPTION** optionList newline |  |
|  |  |  |
| optionList | optionList commaOptNewline optionItem |  |
|  | | optionItem |  |
|  |  |  |
| optionItem | **CASEMAP** **:** mapType |  |
|  | | **LJMP** |  |
|  | | **NOLJMP** |  |
|  | | **LANGUAGE** **:** langType |  |
|  | | **READONLY** |  |
|  | | **NOREADOLY** |  |
|  |  |  |
| mapType | **ALL** |  |
|  | | **NOTPUBLIC** |  |
|  | | **NONE** |  |
|  |  |  |
| ifDir | ifBlock elseifList elseBlock endifStatement |  |
|  |  |  |
| ifBlock | ifStatement inSegDirList |  |
|  |  |  |
| ifStatement | **IF** constExpr newline |  |
|  |  |  |
| elseifList | elseifList elseifBlock |  |
|  | | elseifBlock |  |
|  | | ***empty*** |  |
|  |  |  |
| elseifBlock | elseifStatement inSegDirList |  |
|  |  |  |
| elseifStatement | **ELSEIF** constExpr newline |  |
|  |  |  |
| elseBlock | elseStatement inSegDirList |  |
|  | | ***empty*** |  |
|  |  |  |
| elseStatement | **ELSE** newline |  |
|  |  |  |
| endifStatement | **ENDIF** newline |  |
|  |  |  |
| ppModule | tokenSequenceList | Entry point for pass 2 of the preprocessor. |
|  |  |  |
| ppDir | ppDefineDir |  |
|  | | ppUndefDir |  |
|  | | ppIncludeDir |  |
|  | | ppifDir |  |
|  |  |  |
| ppDefineDir | **#DEFINE** ppMacroId ppMacroParameterList **)** text newline |  |
|  | | **#DEFINE** id text newline |  |
|  |  |  |
| ppMacroId | id**(** | No whitespace between id and “(“. |
|  |  |  |
| ppMacroParameterList | ppMacroParameterList **,** ppMacroParameter |  |
|  | | ppMacroParameter |  |
|  |  |  |
| ppMacroParameter | id |  |
|  |  |  |
| ppUndefDir | **#UNDEF** id newline |  |
|  |  |  |
| ppIncludeDir | **#INCLUDE** fileSpec newline |  |
|  |  |  |
| ppifDir | ppifBlock ppelseifList ppelseBlock ppendifStatement |  |
|  |  |  |
| ppIfBlock | ppifStatement ppTokenSequenceList |  |
|  |  |  |
| ppifStatement | **#IF** constExpr newline |  |
|  |  |  |
| ppelseifList | ppelseifList ppelseifBlock |  |
|  | | ppelseifBlock |  |
|  | | ***empty*** |  |
|  |  |  |
| ppelseifBlock | ppelseifStatement ppTokenSequenceList |  |
|  |  |  |
| ppelseifStatement | **#ELSEIF** constExpr newline |  |
|  |  |  |
| ppelseBlock | ppelseStatement ppTokenSequenceList |  |
|  | | ***empty*** |  |
|  |  |  |
| ppelseStatement | **#ELSE** newline |  |
|  |  |  |
| ppendifStatement | **#ENDIF** newline |  |
|  |  |  |
| ppTokenSequenceList | ppTokenSequenceList ppTokenSequence |  |
|  | | ppTokenSequence |  |
|  | | ***empty*** |  |
|  |  |  |
| ppTokenSequence | ppDir |  |
|  | | ppText newline |  |
|  |  |  |
| ppText | ppCharacter text |  |
|  | | ppCharacter |  |
|  |  |  |
| ppCharacter | Any character except hash (#). |  |
|  |  |  |
| lineDir | **LINE** constExpr fileSpec newline |  |
|  |  |  |
| ppItemList | ppItemList ppItem | Entry point for the pass 1 of the preprocessor. |
|  | | ppItem |  |
|  |  |  |
| ppItem | whiteSpace |  |
|  | | newline |  |
|  | | string |  |
|  | | symbol |  |
|  |  |  |
| symbol | symbol symbolCharacter |  |
|  | | symbolCharacter |  |
|  |  |  |
| symbolCharacter | Any character except whiteSpaceCharacter. |  |