

```
-- file ErrorTab.Mesa
-- last modified by Satterthwaite, July 17, 1978 11:02 AM
```

DIRECTORY

```
ErrorDefs: FROM "errordefs",
SymDefs: FROM "symdefs",
TreeDefs: FROM "treedefs";
```

```
ErrorTab: PROGRAM =
BEGIN
```

```
pname: ARRAY TreeDefs.NodeName[assignx..uparrow] OF STRING =
["←",
" OR ", " AND ", "=", "#", "<", ">=", ">", "<=", " IN ", " ~IN ",
"+", "-", "*", "/", " MOD ",
":", "::", ":",
"~", "~", "@", "↑"];
```

```
fname: ARRAY TreeDefs.NodeName[min..memory] OF STRING =
["MIN", "MAX", "LONG", "ABS",
"SIZE", "FIRST", "LAST", "DESCRIPTOR",
"LENGTH", "BASE", "LOOPHOLE", "REGISTER", "MEMORY"];
```

```
ErrorMessages: ARRAY ErrorDefs.ErrorCode OF STRING = [
```

```
-- general purpose errors (no parameters)
--compilerError,
" FATAL COMPILER ERROR",
--unimplemented,
"unimplemented construct",
--other,
"unspecified error",

-- errors with no parameters
--bodyEntries,
"too many procedure or signal definitions",
--bodyType,
"illegal initialization to body",
--catchLabel,
"misplaced RETRY or CONTINUE",
--descriptor,
"misused DESCRIPTOR",
--elision,
"field cannot be elided",
--exit,
"illegal EXIT or LOOP",
--fieldSize,
"variable or field too large",
--illDefinedReturn,
"default return of anonymous value",
--inlineType,
"illegal initialization to inline",
--interfaceEntries,
"too many interface items",
--keys,
"keywords not permitted",
--long,
"misused LONG",
--misplacedCatch,
"misplaced catch phrase",
--misplacedEntry,
"unmonitored ENTRY or INTERNAL",
--misplacedMonitorRef,
"unmonitored WAIT or NOTIFY",
--misplacedResume,
"illegal RESUME",
--misplacedReturn,
"illegal RETURN",
--misplacedStop,
"illegal STOP",
--missingBinding,
"binding expression required",
--noAccess,
"attempt to access private fields",
--nonOrderedType,
"index or interval type must be ordered",
```

```
--nonTypeCons,
    "unrecognizable type constructor",
--nonVoidStmt,
    "procedure used as a statement returns a value",
--instLength,
    "instruction has incorrect length",
--relative,
    "misused RELATIVE",
--staticNesting,
    "body or catch phrase nested too deeply",
--stringLength,
    "invalid string length",
--subrangeBounds,
    "interval bounds are not constant",
--subrangeNesting,
    "interval bounds are out of range",
--voidExpr,
    "procedure within an expression returns no value",

-- errors mapping hash to sei parameters
--unknownId,
    "is undeclared",
--unknownField,
    "is not valid as a field selector",

-- errors with hash parameters
--ambiguousId,
    "is ambiguous without discrimination",
--duplicateId,
    "is multiply defined",
--unknownTag,
    "is not a legal variant tag",
--duplicateTag,
    "labels multiple variants",
--unknownVariant,
    "does not name a variant",
--unknownKey,
    "is not valid as a key",
--duplicateKey,
    "is repeated as a key",
--omittedKey,
    "cannot be defaulted",
--openId,
    "must label a record",
--fileName,
    "cannot be opened",
--fileVersion,
    "referenced in different versions",
--moduleId,
    "is not a module name",

-- errors with sei parameters
--addressOverflow,
    "exceeds addressing limits",
--circularType,
    "leads to a circular type definition",
--circularValue,
    "has ill-defined or circular value",
--forwardRef,
    "is used before set",
--internalCall,
    "is a misused INTERNAL",
--missingBase,
    "requires a basing value",
--missingLock,
    "required for lock",
--nonDefinition,
    "improperly defined for a DEFINITIONS module",
--nonTagType,
    "has a type illegal for a tag",
--nonTypeId,
    "is not a type",
--notImported,
    "must come from an imported interface",
--recordGap,
    "is followed by a gap",
```

```

--typeLength,
    "has ill-defined or infinite length",
--varExport,
    "has no exportable value",

-- errors with tree parameters
--controlId,
    "is an improper control variable",
--duplicateLabel,
    "is multiply defined",
--exportClash,
    "is exported with incorrect type",
--mixedRepresentation,
    "has signed/unsigned ambiguity",
--noApplication,
    "does not allow an argument list",
--nonAddressable,
    "does not allow indirect reference",
--nonConstant,
    "has nonconstant value",
--nonLHS,
    "cannot be updated",
--nonVariantLabel,
    "specifies an inadmissible test",
--noTarget,
    "requires explicit type",
--notPortable,
    "is not importable/exportable",
--noVariants,
    "has no variants",
--relationType,
    "relational has inadmissible operands",
--sizeClash,
    "has incorrect or ambiguous length",
--spuriousBinding,
    "explicit selection not permitted",
--typeClash,
    "has incorrect type",
--unknownLabel,
    "is undeclared",

-- errors with numeric parameters
--listShort
    "too few elements in list",
--listLong
    "too many elements in list",

-- warnings with no parameters
--attrClash
    "invalid attribute combination",
--pointerInit
    "multiple initialization with a pointer",

-- warnings with sei parameters
--privateExport
    "is private but matches an export",
--unusedExport
    "receives no exported values",
--unusedId
    "is never referenced",
--unusedImport
    "supplies no imported values",

-- warnings with tree parameters
--unsignedCompare
    "will use unsigned comparison"

];

NodePrintName: ARRAY TreeDefs.NodeName OF STRING =
["list", "item",
 "declitem",
 "modeTC", "basicTC", "enumeratedTC",
 "recordTC", "monitoredTC", "variantTC",
 "pointerTC", "arrayTC", "arraydescTC",
 "procTC", "processTC", "portTC", "signalTC", "errorTC", "programTC",

```

```

"definitionTC", "unionTC", "relativeTC", "subrangeTC", "longTC",
"implicitTC", "frameTC", "discrimTC",
"entry",
"unit", "module", "body", "inline", "block", "lambda",
"assign", "construct", "vconstruct", "rowcons", "extract",
"ifstmt", "casestmt", "casetest", "caseswitch", "bindstmt",
"dostmt", "forseq", "upthru", "downthru",
"return", "goto", "exit", "loop",
"syserror", "resume", "continue", "retry", "catchmark",
"restart", "stop", "wait", "notify", "broadcast", "unlock", "nullstmt",
"label", "openstmt", "enable", "catchphrase",
"dst", "lst", "lstf", "procinit",
"apply", "call", "portcall", "signal", "error", "xerror",
"start", "fork", "join",
"index", "dindex", "seqindex", "reloc",
"constructx", "vconstructx", "unionx", "rowconsx",
"ifexp", "caseexp", "bindexp", "assignx", "or", "and",
"relE", "relN", "relL", "relGE", "relG", "relLE", "in", "notin",
"plus", "minus", "times", "div", "mod",
"dot", "cdot", "dollar",
"not", "uminus", "addr", "uparrow",
"min", "max", "lengthen", "abs", "size", "first", "last",
"arraydesc", "length", "base", "loophole",
"register", "memory",
"intOO", "intOC", "intCO", "intCC",
"clit", "llit", "cast", "float", "align", "openexp", "new",
"stringinit", "signalinit", "portinit", "mwconst", "temp",
"none",
"internal", "diritem"];

```

```

TypePrintName: ARRAY SymDefs.TypeClass OF STRING =
  ["mode", "basic", "enumerated", "record", "pointer", "array",
  "arraydesc", "transfer", "definition", "union",
  "relative", "subrange", "long", "real", "NIL"];

```

```

ModePrintName: ARRAY SymDefs.TransferMode OF STRING =
  ["procedure", "port", "signal", "error", "process", "program", "none"];

```

```

END.

```