

Level Procedures 2

*** Compilers 20/12/75

```

    static
    §s
        Ignoring = false
5    §s

    manifest
    §m
        LEVELMAX = 2047
10
        CheckStructureFull[] is
            unless StructureNPtr > StructureLPtr ≤ LEVELMAX do
                Error[2, HARD, DUMMY, Finish]

15    GlobalNo[] = valof
        §GN
            GN := GN + 1
            resultis GN
        §GN
20
        StaticNo[] = valof
        §SN
            SN := SN + 1
            resultis SN
25    §SN

        ManFunNo[] = valof
        §MFN
            MFN := MFN + 1
30    resultis MFN
        §MFN

        IsName[C] = IsCType[C, NAME]

35    Ignoreif[Cond] is
        § Ignoring := Cond
            Send := Cond → NullProgram, Write
        §

40    OutputDef[Name, Type] is
        §OD
            Send[Type]
            Send[Name]
        §OD
45    §m

    let FormLevel[Prev, Type, Gen, NameL] = valof
    §FL
50    let L = StructureVec + StructureLPtr
        StructureLPtr := StructureLPtr + 4
        CheckStructureFull[]
        Copy[lv Prev, L, 4]
        resultis L
55    §FL

    and FormNameBlock[Prev, Type, Gen, N, Val, No, Fld] = valof
    §FNB
60    StructureNPtr := StructureNPtr - 7
        CheckStructureFull[]

```

```

    § let N = StructureVec + StructureNPtr
      Copy[lv Prev, N, 7]
      resultis N
65 $FNB

and AddDef[N, AddType, DefType] be
$AD
70   let F = 0
      switchon AddType into
      §S
        case SIMPLE:
        case GLOBAL:
75     case MANIFEST:
        case MANFN:
          AddName[N, AddType]
        endcase

80     case STATIC:
        case TABLE:
          F := NameBlock[N]
          test NameType[F] = GLOBAL
            ifso UpdateGlobNN[F]
85         ifnot AddName[N, AddType]
          endcase

          default:CompilerError[1001]
      §S
90   OutputDef[N, DefType]
$AD

and UpdateGlobNN[F] be
95   test NameNo[F] = UNDEFINED
      ifso UpdateNameNo[F, GlobalNo[]]
      ifnot Error[23, HARD, NameField[F], NullProgram]

100 and AddName[N, Type] be
      unless Ignoring do
      §AN
        let No = (Type = STATIC ∨ Type = LABEL ∨ Type = TABLE) → StaticNo[],
              (Type = MANFN) → ManFunNo[], UNDEFINED
105   let v = FormNameBlock[NameBlock[N], Type, Generation[PresentLevel],
              NameList[PresentLevel], UNUSED, No, N]
          UpdateNameList[PresentLevel, v]
          UpdateNameBlock[N, v]
      §AN
110
****

```