```
Declarations for Pass 4
 *** Compilers 3/7/76


    manifest GD = 430

    global
 5  §g
        GenerateCode : GD
        SetUpCode    : GD + 1
        ProperVec    : GD + 2
    $g
 10
    static
    §s
        GlobalType    = 0
        GlobalLoc     = 0
 15     GlobalVal     = 0
        StaticType    = 0
        StaticLastUse = 0
        StaticVal     = 0
        MFVal         = 0
 20     MFLastUse     = 0
        StringUse     = 0
        StringData    = 0
        TableData     = 0
        HopList       = 0
 25     BackHopList   = 0
        SwitchList    = 0
        DiagAddr      = 0
        DiagGlobNo    = 0
        DiagName      = 0
 30     DiagString    = 0
        SL            = 0
        TDPtr         = 0
        DN            = 0
        Vector        = 0
 35     Ptr           = 0
        BodyVec       = 0
        NIn           = 0
        CSize         = 0
        FirstCode     = 0
 40     FirstWord     = 0
        LastWord      = 0
        CodeFull      = 0
        SpareByte     = 0
        SNec          = false
 45     ExitNec       = false
        HopNext       = false
        HopPtNext     = false
        FnExitNext    = false
        NextSS        = UNSET
 50     HopNo         = 0
        HopPtNo       = 0
    $s

    manifest
 55 §m
        SetUpBody[Size] is
        §   Vector := ProperVec[Size]
            BodyVec := Vector
        §
 60
        SetCodePtrs[] is
```

```
        §   FirstWord := BodyVec
            LastWord := BodyVec + CodeSize
            FirstCode := LastWord + 1
 65     §

        AlterBody[] is
        §   BodyVec := FirstCode
            Ptr := LastWord - FirstCode + 1
 70     §

        ResetBody[] is
            BodyVec := Vector

 75     ReturnBody[Size] is
            ReturnVec[Vector, Size]

        CodeWord[Loc] = rv Loc
        UpdateCode[Loc, v] is rv Loc := v
 80     UpdateBody[Ptr, v] is BodyVec↓Ptr := v
        CopytoBody[Vec, n] is
        §   Copy[Vec, BodyVec + Ptr, n]
            Ptr := Ptr + n
        §
 85
        CopyStringtoBody[S] is
            CopytoBody[S, LHalf[S↓0]/2 + 1]

        TransferIntoBody[S, p, n] is
 90         TransferIn[S, BodyVec + p, n]

        OutputSection[Type, n] is
        §   Write[Type]
            Write[n]
 95         TransferOut[Output, BodyVec, n]
        §

        HOPSUM       = HOPIFZEsm + HOPIFNZsm
        FIRSTMEDINST = 8260
100     LASTMEDINST  = 8317

        RevCondType[CT] = HOPSUM - CT

        CodePos[] = FirstCode - 1
105     NextCodePos[] = FirstCode
        LastCodePos[] = FirstCode - 2

        WordafterHop[n] = valof
        §   let l = HopList↓n
110         resultis l < 0 → 0, CodeWord[l]
        §

        DiagNo[] = valof
        §   DN := DN + 1
115         resultis DN
        §
    $m


****
```