Eg 5.1

I/ See formulation in question
    Set 2 Q 5.

II/ Read a line of the data file
    Classify the student
    Print result
    Repeat
PROGRAM

! Description
! Declare variables
! Print header:
! Loop until end-of-file.

10 READ line of file (SID, HOURS)
! Classify student

IF ( 0 < HOURS .AND. HOURS <= 30) &
   CLASS = 'freshman'
   IF ( 0 < HOURS <= 30) in FORTRAN
   fails.

   etc

! Print line of output:
PRINT SID, HOURS, CLASS
! End of loop
GO TO 10
END
IV  PROGRAM  student
! Description: classify students by hours.
! Declare variables:
INTEGER :: SID, HOURS
CHARACTER (len=9) :: CLASS
! Print header:
WRITE(*,*) ! Prints blank line.
WRITE(*,*)
WRITE(*,*) 'REGISTRATION REPORT'
WRITE(*,*) 'STUDENT HOURS CLASSIFICATION'
! Open file:
OPEN (1, FILE = 'student.dat')
READ (1, *, END = 20) SID, HOURS
! Classify student:
IF (0 < HOURS .AND. HOURS <= 30) THEN
   CLASS = 'Freshman'
ELSEIF (30 < HOURS .AND. HOURS <= 60) THEN
   CLASS = 'Sophomore'
ELSEIF (60 < HOURS .AND. HOURS <= 90) THEN
   CLASS = 'Junior'
ELSEIF (90 < HOURS) THEN
   CLASS = 'Senior'
END IF
ELSE
  CLASS = 'Unknown'
ENDIF
WRITE (*) SID, HOURS, CLASS
WRITE (30) format (TS, I4, T20, I3, T40, A)
GO TO 10
20 STOP ! Replace by other code for Part II.
END 'End of file reached.'

Alternatives:
I
  [20 STOP
  END
II Replace GO TO loop by a DO loop.
DO ! New statement
  10 READ (1, *, END=20) SID, HOURS
  :

! Go TO 10
ENDDO ! New statement

Do I = 1, 1000  \underline{\text{loop counter}}
  10 READ (1, *, END=20) SID, HOURS
  :

! Go TO 10
ENDDO

! Safer than DO without counter - it only executes 1000 times, at most.
This assumes there are \(< 1000\) students in data file.
Older form:

\[ \text{Do} \quad 40 \quad \text{I} = 1,1000 \]

\[ : \]

\[ 40 \ \text{CONTINUE} \quad ! \text{in place of ENDDO.} \]

New form DO with label:

\[ \text{Loop-name : Do I = 1,1000} \]

\[ \text{ENDDO Loop-name} \]

Named loops are easy to distinguish. Useful with lots of DO loops.
We use a counter for each class of student.

\[
\text{INTEGER :: ifr = 0, iso = 0, iju = 0, ise = 0, iun = 0}
\]

\[
\text{\vdots}
\]

\[
\text{IF } (\text{OXHOURS . AND. HOURS } \leq 30) \text{ THEN}
\]
\[
\text{CLASS = 'Freshman'}
\]
\[
\text{ifr = ifr + 1}
\]

\[
\text{ELSEIF } (30 < \text{HOURS . AND. HOURS } \leq 60) \text{ THEN}
\]
\[
\text{CLASS = 'Sophomore'}
\]
\[
\text{iso = iso + 1}
\]

\[
\text{ELSEIF ...}
\]
\[
\text{\vdots}
\]

\[
\text{ENDIF}
\]

Add code to output results:
Replace the IF-block by a CASE-block.

```
SELECT CASE (HOURS)
    CASE (1:30)
        CLASS = 'Freshman'
    CASE (31:60)
        CLASS = 'Sophomore'
    CASE (61:90)
        CLASS = 'Junior'
    CASE (91:)
        CLASS = 'Senior'
    CASE DEFAULT
        CLASS = 'Unknown'
END SELECT
```