

Appendix 2

KES KNOWLEDGE-BASE CODING

This Appendix contains the computer coding for twelve subject area expert systems. The computer coding for Art is contained in Chapter Five.

- Computing
- Business Education — Typing
- Drama
- English
- Home Economics
- Languages Other Than English — Indonesian, Italian, and Greek
- Mathematics
- Music
- Physical Education & Health
- Science
- Social Education
- Technical Studies

\ NEXT-COMPUTING.

constants:

BANNER:

```

"*****",
" ".

```

HOLD:

```

"           Please press c to continue".

```

%

types:

units:mlt

(CO100, CO200, CO310, CO320, CO330, CO340, CO400, No Computing, No_CO).

%

attributes:

StudSName:str [default: " "].

StudFName:str [default: " "].

StudYear:int.

CurrTerm:int.

NOTES1:str.

NOTES2:str.

NOTES3:str.

NOTES4:str.

TIMING:str.

NoteDTP:truth.

NoteComputing:truth.

NoteMM:truth.

NoteApplications:truth.

DoneUnits:units.

RECOMMEND:units.

COMPUTINGjssc:int [default:0].

COMPUTINGcmplt:int [default: 0].

Initial Interests:sgl

(DTP

```

    {question:" Desktop publishing"},

```

Graphics

```

    {question:" Graphics"},

```

MM

```

    {question:" Multi media"},

```

Applications

```

    {question:" Applications"})

```

```

{question: "Which of these Computing units would you prefer next term?"," " }

```

```

{explain: "Our recommendations take in account your preferences as well as your
grades in the pre-requisite units."}.

```

INTERESTS:sgl

(DTP, Graphics, MM, Applications).

```

FEELINGS:sgl
(Really keen, Enjoy it, Its okay, Prefer not, Definitely not)
{question: "What are you feelings towards doing Computing next term?," " "}
{explain: "We only recommend extension units to those students who wish to study
more Computing!"}.

```

```

INFO:sgl
(Justify
  {question: "Explain the recommendation."},
Again
  {question: "Enter new information about your preferences."},
Continue
  {question: "Continue with the next subject area."})
{question: "What would you like to do now?," " "}.

```

```

%
classes:

```

```

GRADES:
attributes:
RESULTS:sgl (A, B, C, D, E, W, Current Unit, N).
WeightedResults:sgl (Satisfactory, Unsatisfactory).
%
endclass.

```

```

%
rules:

```

```

Setup diagnosis:
  if true
  then  NOTES1 = " ".
        NOTES2 = " ".
        NOTES3 = " ".
        NOTES4 = " ".
        TIMING = " ".
  endif.

```

```

\ *****

```

```

\ Stage One

```

```

CO10 diagnosis:
\ If student has not completed any computing units
\ and feelings are Really keen, Enjoy it, or Its okay
\ then recommend the introductory unit.
  if  COMPUTINGcmplt = 0
    and FEELINGS = Really keen | Enjoy it | Its okay
  then
    RECOMMEND = CO100.
  endif.

```

```

\ *****
\      Stage Two

CO20 diagnosis:
\ If student has completed the introductory unit
\ and feelings are Really keen, Enjoy it, or Its okay
\ then recommend the Stage Two unit.
    if
    DoneUnits = CO100
    and DoneUnits # CO200
    and FEELINGS = Really keen | Enjoy it | Its okay
    then
    RECOMMEND = CO200.
    endif.

\ *****

\      Stage Three

CO31 diagnosis:
\ If student has a satisfactory result in the Stage Two unit
\ and is interested in Desk Top Publishing
\ then recommend the Stage Three unit.
    if
    GRADES:CO200>WeightedResults = Satisfactory
    and INTERESTS = DTP
    and DoneUnits # CO310
    then
    RECOMMEND = CO310.
    endif.

CO32 diagnosis:
\ If student has a satisfactory result in the Stage Two unit
\ and is interested in Graphics
\ then recommend the Stage Three unit.
    if
    GRADES:CO200>WeightedResults = Satisfactory
    and INTERESTS = Graphics
    and DoneUnits # CO320
    then
    RECOMMEND = CO320.
    endif.

CO33 diagnosis:
\ If student has a satisfactory result in the Stage Two unit
\ and is interested in Multi Media
\ then recommend the Stage Three unit.
    if
    GRADES:CO200>WeightedResults = Satisfactory
    and INTERESTS = MM
    and DoneUnits # CO330
    then
    RECOMMEND = CO330.
    endif.

```

CO34 diagnosis:

```
\ If student has a satisfactory result in the Stage Two unit
\ and is interested in Applications
\ then recommend the Stage Three unit.
  if
    GRADES:CO200>WeightedResults = Satisfactory
    and INTERESTS = Applications
    and DoneUnits # CO340
  then
    RECOMMEND = CO340.
  endif.
```

CO3_20 diagnosis:

```
\ Situation when the results in the Stage Two unit were unsatisfactory
\ but student is interested in further computing
\ then recommend repeating the Stage Two unit.
  if
    GRADES:CO200>WeightedResults = Unsatisfactory
    and FEELINGS = Really keen | Enjoy it | Its okay
    and DoneUnits # CO310 | CO320 | CO330 | CO340
  then
    RECOMMEND = CO200.
  reassert NOTES1 = "*** You wanted to do more Computing".
  reassert NOTES2 = " but your grade in CO200 was unsatisfactory".
  reassert NOTES3 = " You may repeat this unit".
  reassert NOTES4 = " or use option 2 below and change to no-computing".
  endif.
```

\ *****

\ Stage Four

CO40 diagnosis:

```
\ If student has a satisfactory result in specific Stage Three units
\ and is interested in Desk Top Publishing or graphics
\ then recommend the Stage Four unit.
  if
    DoneUnits = CO310 | CO320
    and INTERESTS = DTP | Graphics
    and DoneUnits # CO400
    and GRADES:CO310>WeightedResults = Satisfactory
    or GRADES:CO320>WeightedResults = Satisfactory
  then
    RECOMMEND = CO400.
  endif.
```

CO40_31 diagnosis:

```
\ If student has completed Stage Three and Four Desk Top Publishing
\ but is interested in more Desk Top Publishing or Graphics
\ and has not done Stage Three Graphics
\ then recommend Stage Three Graphics and explain this.
  if
    DoneUnits = CO320 & CO400
    and INTERESTS = DTP | Graphics
    and DoneUnits # CO310
      then
        RECOMMEND = CO310<0.50>.
        NoteDTP = true.
      endif.
```

CO40_32 diagnosis:

```
\ If student has completed Stage Three and Four Graphics
\ but is interested in more Desk Top Publishing or Graphics
\ and has not done Stage Three Desk Top Publishing
\ then recommend Stage Three Desk Top Publishing and explain this.
  if
    DoneUnits = CO310 & CO400
    and INTERESTS = DTP | Graphics
    and DoneUnits # CO320
      then
        RECOMMEND = CO320<0.50>.
        NoteDTP = true.
      endif.
```

CO4_3 diagnosis:

```
\ Situation when the results in the Stage Three unit were unsatisfactory
\ but student is interested in further computing
\ then recommend repeating the Stage Three unit
\ or select another computing topic at Stage Three.
  if
    GRADES:CO310>WeightedResults = Unsatisfactory
    or GRADES:CO320>WeightedResults = Unsatisfactory
      then
        RECOMMEND = CO310 | CO320.
reassert NOTES1 = "*** You wanted to do more Graphics or Desk Top Publishing".
reassert NOTES2 = " but your grade in CO310 or CO320 was unsatisfactory".
reassert NOTES3 = " You may repeat this unit or try the other".
reassert NOTES4 = " or use option 2 below and change to no-computing".
      endif.
```

\ *****

\ Student feelings towards and interests in non-compulsory units

Computing_Prefer_Not response:

```
  if
    FEELINGS = Prefer not
      then
        RECOMMEND = No_CO <0.50>.
      endif.
```

Computing_Definitely_Not response:

```
if
  FEELINGS = Definitely not
  then
  RECOMMEND = No_CO <1.00>.
endif.
```

DTP_keen response:

```
if
  FEELINGS = Really keen
  and Initial Interests = DTP
  then
  INTERESTS = DTP <1.00>.
endif.
```

DTP_enjoy response:

```
if
  FEELINGS = Enjoy it
  and Initial Interests = DTP
  then
  INTERESTS = DTP <0.75>.
endif.
```

DTP_OK response:

```
if
  FEELINGS = Its okay
  and Initial Interests = DTP
  then
  INTERESTS = DTP <0.30>.
endif.
```

Graphics_keen response:

```
if
  FEELINGS = Really keen
  and Initial Interests = Graphics
  then
  INTERESTS = Graphics <1.00>.
endif.
```

Graphics_Enjoy response:

```
if
  FEELINGS = Enjoy it
  and Initial Interests = Graphics
  then
  INTERESTS = Graphics <0.75>.
endif.
```

Graphics_OK response:

```
if
  FEELINGS = Its okay
  and Initial Interests = Graphics
  then
  INTERESTS = Graphics <0.30>.
endif.
```

MM_keen response:

```
if
  FEELINGS = Really keen
  and Initial Interests = MM
  then
    INTERESTS = MM <1.00>.
  endif.
```

MM_Enjoy response:

```
if
  FEELINGS = Enjoy it
  and Initial Interests = MM
  then
    INTERESTS = MM <0.75>.
  endif.
```

MM_OK response:

```
if
  FEELINGS = Its okay
  and Initial Interests = MM
  then
    INTERESTS = MM <0.30>.
  endif.
```

Applications_keen response:

```
if
  FEELINGS = Really keen
  and Initial Interests = Applications
  then
    INTERESTS = Applications <1.00>.
  endif.
```

Applications_Enjoy response:

```
if
  FEELINGS = Enjoy it
  and Initial Interests = Applications
  then
    INTERESTS = Applications <0.75>.
  endif.
```

Applications_OK response:

```
if
  FEELINGS = Its okay
  and Initial Interests = Applications
  then
    INTERESTS = Applications <0.30>.
  endif.
```



```

\ *****
\   Genre completed

CO40_t diagnosis:
  if
    DoneUnits = CO400 & CO310 & CO320
    and INTERESTS = DTP | Graphics
    then
      RECOMMEND = No_CO <0.50>.
      NoteComputing = true.
    endif.
CO33_t diagnosis:
  if
    DoneUnits = CO330
    and INTERESTS = MM
    then
      RECOMMEND = No_CO <0.50>.
      NoteMM = true.
    endif.
CO34_t diagnosis:
  if
    DoneUnits = CO340
    and INTERESTS = Applications
    then
      RECOMMEND = No_CO <0.50>.
      NoteApplications = true.
    endif.

\ *****
\   Output messages

NDTP details:
  if
    NoteDTP = true
    then
      reassert NOTES1 = "*** You wanted to do more DTP and Graphics".
      reassert NOTES2 = " and as you have completed CO400".
      reassert NOTES3 = " we recommend you enrol in another unit. ***".
    endif.
NC details:
  if
    NoteComputing = true
    then
      reassert NOTES1 = "*** You wanted to do more DTP and Graphics".
      reassert NOTES2 = " but you have completed all eligible units".
      reassert NOTES3 = " You may like to examine other topics?. ***".
    endif.
NMM details:
  if
    NoteMM = true
    then
      reassert NOTES1 = "*** You wanted to do more Multi-media".
      reassert NOTES2 = " but you have completed all eligible units".
      reassert NOTES3 = " You may like to examine other topics?. ***".
    endif.

```

NApplications details:

```

if
  NoteApplications = true
then
  reassert NOTES1 = "*** You wanted to do more Applications".
  reassert NOTES2 = " but you have completed all eligible units".
  reassert NOTES3 = " You may like to examine other topics?. ***".
endif.

```

Extension diagnosis:

```

if    COMPUTINGcmplt lt 4
then  reassert TIMING = "Extension".
endif.

```

Specialisation diagnosis:

```

if    COMPUTINGcmplt ge 4
then  reassert TIMING = "Specialisation".
endif.

```

\

%

actions:

```

read "KES-STU-DATA-CO", StudSName, StudFName, StudYear, CurrTerm,
GRADES, GRADES(RESULTS), DoneUnits.

```

\ Interpretation of raw data from student CBASS records.

```

forall G:GRADES do
  while G>RESULTS = N do
    erase G>RESULTS.
    message BANNER.
  message "Our recommendations take into account your previous units and grades".
  message "We already know your past results, but not this term's".
  message "Please enter the grade you reasonably expect for COMPUTING this term".
  askfor G>RESULTS.
  endwhile.
  if G>RESULTS = A
    then G>WeightedResults = Satisfactory <1.00>.    endif.
  if G>RESULTS = B
    then G>WeightedResults = Satisfactory <0.85>.    endif.
  if G>RESULTS = C
    then G>WeightedResults = Satisfactory <0.70>.    endif.
  if G>RESULTS = D
    then G>WeightedResults = Satisfactory <0.50>.    endif.
  if G>RESULTS = E
    then G>WeightedResults = Unsatisfactory <0.50>.  endif.
  if G>RESULTS = W
    then G>WeightedResults = Satisfactory <0.10>.    endif.
endforall.

```

```

\
*****

\      Counting units satisfactorily completed
\      There are no jssc requirements

      COMPUTINGcmplt = 0.

      forall G:GRADES do
        if G>WeightedResults = Satisfactory
          then reassert COMPUTINGcmplt = COMPUTINGcmplt + 1.
          endif.
        endforall.

      if DoneUnits = No Computing
        then reassert COMPUTINGcmplt = COMPUTINGcmplt - 1.
        endif.

\
*****

      obtain RECOMMEND.

\
*****

\      Screen output.

      message BANNER.
      message combine ("      Computing recommendation summary - ", StudFName).
      message " ", "Units thus far ", DoneUnits, " ".
      \message combine ("jssc ",COMPUTINGjssc," passed ", COMPUTINGcmplt).

      if DoneUnits = CO100
        then
          message combine ("Interest in Computing - ", FEELINGS).
          endif.
      if RECOMMEND # No_CO and DoneUnits = CO100
        then
          message combine ("Requested units - ", Initial Interests).
          endif.

      if NOTES1 # " "
      then  message NOTES1.
           message NOTES2.
           message NOTES3.
           message NOTES4, " ". endif.

      message combine ("We recommend that you apply for ", RECOMMEND).
      message " ".
      message combine ("This recommendation is for ",TIMING," unit/s.").

\

```

```
*****
\      File output.
write "RcmdCO", TIMING, RECOMMEND.
\ *****
\      Termination
if INFO = Justify      then justify RECOMMEND.
                        message HOLD.
                        stop.
                        endif.
if INFO = Again       then nextcase.
if INFO = Continue    then stop.
                        endif.
%

```

\ NEXT-TYPING.

constants:

BANNER:

```

"*****",
" "

```

HOLD:

```

"           Please press c to continue".

```

%

types:

units:mlt

```

(BE110, BE210, BE220, BE311, BE312, BE322, No Typing, No Business Ed,
No_BE).

```

%

attributes:

StudSName:str [default: " "].

StudFName:str [default: " "].

StudYear:int.

CurrTerm:int.

NOTES1:str.

NOTES2:str.

NOTES3:str.

TIMING:str.

NotePersonal:truth [default: false].

NoteCareer:truth [default: false].

NoteWord:truth [default: false].

DoneUnits:units.

BEjssc:int [default:0].

TYPINGcmplt:int [default: 0].

RECOMMEND:units.

Initial Interests:sgl

(Personal

```

    {question:" Typing for Personal use"},

```

Career

```

    {question:" Typing as a possible career"},

```

Word

```

    {question:" Word processing"})

```

```

{question: "Which of these Typing units would you prefer next term?"," " }

```

```

{explain: "Our recommendations take in account your preferences as well as your
grades in the pre-requisite units."}.

```

INTERESTS:sgl

(Personal, Career, Word).

```

TypingFEELINGS:sgl
(Really keen, Enjoy it, Its okay, Prefer not, Definitely not)
{question: "What are you feelings towards doing Typing next term?"," "}
{explain: "We only recommend extension units to those students who wish to study
more Typing!"}.

```

```

INFO:sgl
(Justify
    {question: "Explain the recommendation."},
Again
    {question: "Enter new information about your preferences."},
Continue
    {question: "Continue with the next subject area."})
{question: "What would you like to do now?"," "}.

```

```

%
classes:

```

```

GRADES:
attributes:
RESULTS:sgl (A, B, C, D, E, W, N).
WeightedResults:sgl (Satisfactory, Unsatisfactory).
%
endclass.

```

```

%
rules:

```

```

Setup output:
    if true
    then  NOTES1 = " ".
          NOTES2 = " ".
          NOTES3 = " ".
          TIMING = " ".
    endif.

```

```

\ *****

```

```

Typing_PreferNot response:
    if
    TYPINGcmplt ge 1
    and TypingFEELINGS = Prefer not
    then
    RECOMMEND = No_BE <0.50>.
    endif.

```

```

Typing_DefinitelyNot response:
    if
    TYPINGcmplt ge 1
    and TypingFEELINGS = Definitely not
    then
    RECOMMEND = No_BE <1.00>.
    endif.

```

Personal_keen response:

```

if
  TypingFEELINGS = Really keen
  and Initial Interests = Personal
then
  INTERESTS = Personal <1.00>.
endif.

```

Personal_enjoy response:

```

if
  TypingFEELINGS = Enjoy it
  and Initial Interests = Personal
then
  INTERESTS = Personal <0.75>.
endif.

```

Personal_OK response:

```

if
  TypingFEELINGS = Its okay
  and Initial Interests = Personal
then
  INTERESTS = Personal <0.30>.
endif.

```

Career_keen response:

```

if
  TypingFEELINGS = Really keen
  and Initial Interests = Career
then
  INTERESTS = Career <1.00>.
endif.

```

Career_Enjoy response:

```

if
  TypingFEELINGS = Enjoy it
  and Initial Interests = Career
then
  INTERESTS = Career <0.75>.
endif.

```

Career_OK response:

```

if
  TypingFEELINGS = Its okay
  and Initial Interests = Career
then
  INTERESTS = Career <0.30>.
endif.

```

Word_keen response:

```

if
  TypingFEELINGS = Really keen
  and Initial Interests = Word
then
  INTERESTS = Word <1.00>.
endif.

```

Word_Enjoy response:

```

if
  TypingFEELINGS = Enjoy it
  and Initial Interests = Word
then
  INTERESTS = Word <0.75>.
endif.

```

Word_OK response:

```

if
  TypingFEELINGS = Its okay
  and Initial Interests = Word
then
  INTERESTS = Word <0.30>.
endif.

```

\ *****

B110 diagnosis:

```

\ If a student has not done BE110
\ then recommend this compulsory unit.
if
  DoneUnits # BE110
then
  RECOMMEND = BE110.
endif.

```

B210 diagnosis:

```

\ If a student has done the introductory unit BE110
\ and wants to do more typing
\ then recommend BE210
if
  DoneUnits = BE110
  and TypingFEELINGS = Really keen | Enjoy it | Its okay
  and DoneUnits # BE210
then
  RECOMMEND = BE210.
endif.

```

B311 diagnosis:

```

\ If a student has done the Stage Two unit
\ and is interested in Personal typing
\ then recommend BE311
if
  DoneUnits = BE210
  and INTERESTS = Personal
  and DoneUnits # BE311
then
  RECOMMEND = BE311.
endif.

```

B312 diagnosis:

```

\ If a student has done the Stage Two unit
\ and is interested in typing as a career
\ then recommend BE312
if
  DoneUnits = BE210
  and INTERESTS = Career
  and DoneUnits # BE312
then
  RECOMMEND = BE312.
endif.

```



```

B322 diagnosis:
\ If a student has done the Stage Two unit
\ and is interested in Word Processing
\ then recommend BE322
    if
    DoneUnits = BE210
    and INTERESTS = Word
    and DoneUnits # BE322
    then
    RECOMMEND = BE322.
    endif.
\ *****
\ terminate genre

done_311:
    if
    DoneUnits = BE311
    and INTERESTS = Personal
    then
    RECOMMEND = No_BE <0.50>.
    reassert NOTES1 = "*** You wanted to do more Personal typing".
    reassert NOTES2 = " but you have completed BE311".
    reassert NOTES3 = " Please use option 2 below for alternatives. ***".
    endif.

done_312:
    if
    DoneUnits = BE312
    and INTERESTS = Word
    then
    RECOMMEND = No_BE <0.50>.
    reassert NOTES1 = "*** You wanted to do more Career preparation".
    reassert NOTES2 = " but you have completed all BE312".
    reassert NOTES3 = " Please use option 2 below for alternatives. ***".
    endif.

done_322:
    if
    DoneUnits = BE322
    and INTERESTS = Word
    then
    RECOMMEND = No_BE <0.50>.
    reassert NOTES1 = "*** You wanted to do more Word Processing".
    reassert NOTES2 = " but you have completed BE322".
    reassert NOTES3 = " Please use option 2 below for alternatives. ***".
    endif.

\ *****
\ timing

Compulsory diagnosis:
    if TYPINGcmplt lt 1
    then reassert TIMING = "Important and Compulsory".
    endif.

Extension diagnosis:
    if TYPINGcmplt = 1
    then reassert TIMING = "Extension".
    endif.

```

```

Specialisation diagnosis:
    if      TYPINGcmlpt ge 2
    then    reassert TIMING = "Specialisation".
    endif.

\ *****

%
actions:

read "KES-STU-DATA-BE", StudSName, StudFName, StudYear, CurrTerm,
GRADES, GRADES(RESULTS), DoneUnits.

\ *****

\      Interpretation of raw data from student CBASS records.

forall G:GRADES do
    while G>RESULTS = N do
        erase G>RESULTS.
        message BANNER.
    message "Our recommendations take into account your previous units and grades".
    message "We already know your past results, but not this term's".
    message "Please enter the grade you reasonably expect for Typing this term".
        askfor G>RESULTS.
    endwhile.
        if      G>RESULTS = A
        then    G>WeightedResults = Satisfactory <1.00>.
        endif.
        if      G>RESULTS = B
        then    G>WeightedResults = Satisfactory <0.85>.
        endif.
        if      G>RESULTS = C
        then    G>WeightedResults = Satisfactory <0.50>.
        endif.
        if      G>RESULTS = D
        then    G>WeightedResults = Unsatisfactory <0.25>.
        endif.
        if      G>RESULTS = E
        then    G>WeightedResults = Unsatisfactory <1.00>.
        endif.
        if      G>RESULTS = W
        then    G>WeightedResults = Satisfactory <0.05>.
        endif.
    endforall.

\ Count units completed

TYPINGcmlpt = 0.
BEjssc = 0.

    forall G:GRADES do
        if      G> WeightedResults = Satisfactory | Unsatisfactory
        then    reassert TYPINGcmlpt = TYPINGcmlpt + 1.
        endif.
    endforall.

```

```

if DoneUnits = No Typing
then reassert TYPINGcmplt = TYPINGcmplt - 1.
endif.

```

```

if DoneUnits = BE110
then reassert BEjssc = BEjssc + 1.
endif.

```

```

\ *****

```

```

obtain RECOMMEND.

```

```

\ *****

```

```

\ Screen output.

```

```

message BANNER.
message combine (" Typing recommendation summary - ", StudFName).
message "Units thus far ", DoneUnits, " ".

```

```

if NOTES1 # " "
then message NOTES1.
      message NOTES2.
      message NOTES3, " ". endif.

```

```

\message combine ("BEjssc", BEjssc, "Typingcmplt ", TYPINGcmplt).

```

```

message combine ("We recommend that you apply for ", RECOMMEND).
message " ".
message combine ("This recommendation is for ", TIMING, " unit/s.").

```

```

\ *****

```

```

\ File output.

```

```

write "RcmdBE", TIMING, RECOMMEND.

```

```

\ *****

```

```

\ Termination

```

```

if INFO = Justify      then justify RECOMMEND.
                        message HOLD.
                        stop.
                        endif.
if INFO = Again       then nextcase.
                        endif.
if INFO = Continue    then stop.
                        endif.

```

```

%

```

\ NEXT-DRAMA.

constants:

BANNER:

```

"*****",
" "

```

HOLD:

```

"           Please press c to continue".

```

%

types:

units:mlt

```

(DR110, DR120, DR130, DR140, DR210, DR220, DR230, DR240, DR310, DR320,
DR330, DR331, DR340, No Drama, No_DR, MS110, MS120, MS210, MS220,
MS230, MS240, MS250, MS310, MS311, MS320, MS321, MS330, MS410, No
Music, MSCM, MSIL, MS1L, MS245).

```

%

attributes:

StudSName:str [default: " "].

StudFName:str [default: " "].

StudYear:int.

CurrTerm:int.

NOTES1:str.

NOTES2:str.

NOTES3:str.

NOTES4:str.

TIMING:str.

NotePerformance:truth.

NoteScript:truth.

NoteDesign:truth.

NoteDance:truth.

DoneUnits:units.

RECOMMEND:units.

DRAMAjssc:int [default:0].

DRAMAcmplt:int [default: 0].

Initial Interests:sgl

(Performance

```

    {question:" Group and individual performances"},

```

Script

```

    {question:" Script writing"},

```

Design

```

    {question:" Stage design and construction"},

```

Dance

```

    {question:" Dance and expressive movement"})

```

```

{question: "Which of these Drama groups would you prefer next term?"," " }

```

```

{explain: "Our recommendations take in account your preferences as well as your
grades in the pre-requisite units."}.

```

INTERESTS:sgl
(Performance, Script, Design, Dance).

FEELINGS:sgl
(Really keen, Enjoy it, Its okay, Prefer not, Definitely not)
{question: "What are you feelings towards doing Drama next term?"," "}
{explain: "We only recommend extension units to those students who wish to study more Drama!"}.

STAGE:sgl
(Stage one, Stage two, Stage three).

INFO:sgl
(Justify
 {question: "Explain the recommendation."},
Again
 {question: "Enter new information about your preferences."},
Continue
 {question: "Continue with the next subject area."})
{question: "What would you like to do now?"," "}

%
classes:

GRADES:
attributes:
RESULTS:sgl (A, B, C, D, E, W, Current Unit, N).
WeightedResults:sgl (Satisfactory, Unsatisfactory).
%
endclass.

%
rules:

Setup diagnosis:
 if true
 then NOTES1 = " "
 NOTES2 = " "
 NOTES3 = " "
 NOTES4 = " "
 TIMING = " "
 STAGE = Stage one.
 endif.

\ *****

\ Student feelings towards and interests in Drama units

Drama_Prefer_Not response:
 if
 DRAMAcmplt ge 1
 and DRAMAJssc ge 3
 and FEELINGS = Prefer not
 then
 RECOMMEND = No_DR <0.50>.
 endif.

Drama_Definitely_Not response:

```

if
  DRAMAcplt ge 1
  and DRAMAJssc ge 3
  and FEELINGS = Definitely not
then
  RECOMMEND = No_DR <1.00>.
endif.

```

Perform_keen response:

```

if
  FEELINGS = Really keen
  and Initial Interests = Performance
then
  INTERESTS = Performance <1.00>.
endif.

```

Perform_Enjoy response:

```

if
  FEELINGS = Enjoy it
  and Initial Interests = Performance
then
  INTERESTS = Performance <0.75>.
endif.

```

Performance_OK response:

```

if
  FEELINGS = Its okay
  and Initial Interests = Performance
then
  INTERESTS = Performance <0.30>.
endif.

```

Perform_negative response:

```

if
  DRAMAJssc lt 3
  and FEELINGS = Prefer not | Definitely not
  and Initial Interests = Performance
then
  INTERESTS = Performance <0.10>.
endif.

```

Script_keen response:

```

if
  FEELINGS = Really keen
  and Initial Interests = Script
then
  INTERESTS = Script <1.00>.
endif.

```

Script_Enjoy response:

```

if
  FEELINGS = Enjoy it
  and Initial Interests = Script
then
  INTERESTS = Script <0.75>.
endif.

```

Script_OK response:

```
if
  FEELINGS = Its okay
  and Initial Interests = Script
  then
    INTERESTS = Script <0.30>.
  endif.
```

Script_negative response:

```
if
  DRAMAJSSC lt 3
  and FEELINGS = Prefer not | Definitely not
  and Initial Interests = Script
  then
    INTERESTS = Script <0.10>.
  endif.
```

Design_keen response:

```
if
  FEELINGS = Really keen
  and Initial Interests = Design
  then
    INTERESTS = Design <1.00>.
  endif.
```

Design_Enjoy response:

```
if
  FEELINGS = Enjoy it
  and Initial Interests = Design
  then
    INTERESTS = Design <0.75>.
  endif.
```

Design_OK response:

```
if
  FEELINGS = Its okay
  and Initial Interests = Design
  then
    INTERESTS = Design <0.30>.
  endif.
```

Design_negative response:

```
if
  DRAMAJSSC lt 3
  and FEELINGS = Prefer not | Definitely not
  and Initial Interests = Design
  then
    INTERESTS = Design <0.10>.
  endif.
```

Dance_keen response:

```
if
  FEELINGS = Really keen
  and Initial Interests = Dance
  then
    INTERESTS = Dance <1.00>.
  endif.
```

Dance_Enjoy response:

```

if
  FEELINGS = Enjoy it
  and Initial Interests = Dance
then
  INTERESTS = Dance <0.75>.
endif.

```

Dance_OK response:

```

if
  FEELINGS = Its okay
  and Initial Interests = Dance
then
  INTERESTS = Dance <0.30>.
endif.

```

Dance_negative response:

```

if
  DRAMAJSSC lt 3
  and FEELINGS = Prefer not | Definitely not
  and Initial Interests = Dance
then
  INTERESTS = Dance <0.10>.
endif.

```

```

\ *****
\      Stage One

```

DR11 diagnosis:

```

\ Students are required to do at least one Drama unit
\ if they are interested in Stage concepts then recommend DR110

```

```

if
  DRAMACMPLT = 0
  and Initial Interests = Performance | Script | Design
then
  RECOMMEND = DR110.
endif.

```

DR_14_11 diagnosis:

```

\ If a student has already done the introductory 'dance' unit DR140
\ if they are interested in 'performance' then recommend DR110

```

```

if
  DoneUnits = DR140
  and Initial Interests = Performance | Script | Design
  and DoneUnits # DR110
then
  RECOMMEND = DR110.
endif.

```

DR14 diagnosis:

```

\ Students are required to do at least one Drama unit
\ if they are interested in Dance then recommend DR140

```

```

if
  DRAMACMPLT = 0
  and Initial Interests = Dance
then
  RECOMMEND = DR140.
endif.

```


DR_11_14 diagnosis:

\ If a student has already done the introductory 'performing' unit DR110

\ if they are interested in Dance then recommend DR140

if

DoneUnits = DR110

and FEELINGS # Prefer not | Definitely not

and Initial Interests = Dance

and DoneUnits # DR140

then

RECOMMEND = DR140.

endif.

\ *****

\ Stage Two

DR21 diagnosis:

\ Students who have completed the introductory Stage unit DR110

\ and are interested in further Performing units

\ are recommended to DR210.

if

DoneUnits = DR110 | DR120 | DR130

and DoneUnits # DR210

and FEELINGS # Prefer not | Definitely not

and INTERESTS = Performance

then

RECOMMEND = DR210.

endif.

DR22 diagnosis:

\ Students who have completed the introductory Stage unit DR110

\ and are interested in further work with Scripts

\ are recommended to DR220.

if

DoneUnits = DR110 | DR120 | DR130

and DoneUnits # DR220

and FEELINGS # Prefer not | Definitely not

and INTERESTS = Script

then

RECOMMEND = DR220.

endif.

DR23 diagnosis:

\ Students who have completed the introductory Stage unit DR110

\ and are interested in further work in stage design

\ are recommended to DR230.

if

DoneUnits = DR110 | DR120 | DR130

and DoneUnits # DR230

and FEELINGS # Prefer not | Definitely not

and INTERESTS = Design

then

RECOMMEND = DR230.

endif.

DR24 diagnosis:

\ Students who have completed the introductory Dance unit DR140
 \ and are interested in further Dance units
 \ are recommended to DR240.

```

    if
        DoneUnits = DR140
        and FEELINGS # Prefer not | Definitely not
        and INTERESTS = Dance
        and DoneUnits # DR240
    then
        RECOMMEND = DR240.
    endif.
  
```

\ *****

\ Stage Three

DR31 diagnosis:

\ Students who have good results in the Performance unit DR210
 \ and are interested in further Performing units
 \ are recommended to DR310.

```

    if
        GRADES:DR210>WeightedResults = Satisfactory
        and INTERESTS = Performance
        and DoneUnits # DR310
    then
        RECOMMEND = DR310.
    endif.
  
```

DR31_22 diagnosis:

\ Students who did not get an A or B in the Performance unit DR210
 \ and are interested in further Performance units
 \ are recommended to another stage two units rather than stage three
 \ with a message to explain this.

```

    if
        GRADES:DR210>WeightedResults = Unsatisfactory
        and INTERESTS = Performance
        and DoneUnits # DR220
    then
        RECOMMEND = DR220<0.50>.
        NotePerformance = true.
    endif.
  
```

DR31_23 diagnosis:

\ Students who did not get an A or B in the Performance unit DR210
 \ and are interested in further Performance units
 \ are recommended to another stage two units rather than stage three
 \ with a message to explain this.

```

    if
        GRADES:DR210>WeightedResults = Unsatisfactory
        and INTERESTS = Performance
        and DoneUnits # DR230
    then
        RECOMMEND = DR230<0.50>.
        NotePerformance = true.
    endif.
  
```

DR31_32 diagnosis:

\ Students who achieved satisfactory grades in the Performance unit DR310
 \ and are interested in further Performance units
 \ are recommended to another stage three units.

```

if
    GRADES:DR310>WeightedResults = Satisfactory
    and INTERESTS = Performance
    and DoneUnits # DR320
then
    RECOMMEND = DR320<0.75>.
    NotePerformance = true.
endif.
  
```

DR31_33 diagnosis:

\ Students who achieved satisfactory grades in the Performance unit DR310
 \ and are interested in further Performance units
 \ are recommended to another stage three units.

```

if
    GRADES:DR310>WeightedResults = Satisfactory
    and INTERESTS = Performance
    and DoneUnits # DR330
then
    RECOMMEND = DR330<0.75>.
    NotePerformance = true.
endif.
  
```

DR32 diagnosis:

\ Students who have good results in the Script unit DR220
 \ and are interested in further work with scripts
 \ are recommended to DR320.

```

if
    GRADES:DR220>WeightedResults = Satisfactory
    and INTERESTS = Script
    and DoneUnits # DR320
then
    RECOMMEND = DR320.
endif.
  
```

DR32_21 diagnosis:

\ Students who did not get an A or B in the Script unit DR220
 \ and are interested in further Script units
 \ are recommended to another stage two units rather than stage three
 \ with a message to explain this.

```

if
    GRADES:DR220>WeightedResults = Unsatisfactory
    and INTERESTS = Script
    and DoneUnits # DR210
then
    RECOMMEND = DR210<0.50>.
    NoteScript = true.
endif.
  
```

DR32_31 diagnosis:

\ Students who achieved satisfactory grades in the Performance unit DR310
 \ and are interested in further Performance units
 \ are recommended to another stage three units.

```

if
  DoneUnits = DR320
  and INTERESTS = Script
  and GRADES:DR320>WeightedResults = Satisfactory
  and DoneUnits # DR310
  then
    RECOMMEND = DR310<0.75>.
    NoteScript = true.
  endif.

```

DR32_23 diagnosis:

\ Students who did not get an A or B in the Script unit DR220
 \ and are interested in further Performance units
 \ are recommended to another stage two units rather than stage three
 \ with a message to explain this.

```

if
  GRADES:DR220>WeightedResults = Unsatisfactory
  and INTERESTS = Script
  and DoneUnits # DR230
  then
    RECOMMEND = DR230<0.50>.
    NoteScript = true.
  endif.

```

DR32_33 diagnosis:

\ Students who achieved satisfactory grades in the Script unit DR330
 \ and are interested in further Performance units
 \ are recommended to another stage three units.

```

if
  GRADES:DR320>WeightedResults = Satisfactory
  and INTERESTS = Script
  and DoneUnits # DR330
  then
    RECOMMEND = DR330<0.75>.
    NoteScript = true.
  endif.

```

DR33 diagnosis:

\ Students who have good results in the Design unit DR230
 \ and are interested in further work in stage design
 \ are recommended to DR330.

```

if
  GRADES:DR230>WeightedResults = Satisfactory
  and INTERESTS = Design
  and DoneUnits # DR330
  then
    RECOMMEND = DR330.
  endif.

```

DR33_21 diagnosis:

\ Students who did not get an A or B in the Design unit DR230
 \ and are interested in further Performance units
 \ are recommended to another stage two units rather than stage three
 \ with a message to explain this.

```

if
    GRADES:DR230>WeightedResults = Unsatisfactory
    and INTERESTS = Design
    and DoneUnits # DR210
then
    RECOMMEND = DR210<0.50>.
    NoteDesign = true.
endif.
  
```

DR33_22 diagnosis:

\ Students who did not get an A or B in the Design unit DR230
 \ and are interested in further Performance units
 \ are recommended to another stage two units rather than stage three
 \ with a message to explain this.

```

if
    GRADES:DR230>WeightedResults = Unsatisfactory
    and INTERESTS = Design
    and DoneUnits # DR220
then
    RECOMMEND = DR220<0.50>.
    NoteDesign = true.
endif.
  
```

DR33_31 diagnosis:

\ Students who achieved satisfactory grades in the Design unit DR330
 \ and are interested in further Performance units
 \ are recommended to another stage three units.

```

if
    GRADES:DR330>WeightedResults = Satisfactory
    and INTERESTS = Design
    and DoneUnits # DR310
then
    RECOMMEND = DR310<0.75>.
    NoteDesign = true.
endif.
  
```

DR33_32 diagnosis:

\ Students who achieved satisfactory grades in the Design unit DR330
 \ and are interested in further Performance units
 \ are recommended to another stage three units.

```

if
    GRADES:DR330>WeightedResults = Satisfactory
    and INTERESTS = Design
    and DoneUnits # DR320
then
    RECOMMEND = DR320<0.75>.
    NoteDesign = true.
endif.
  
```

DR3_14 diagnosis:

```

if
  STAGE = Stage three
  and INTERESTS = Script
  and DoneUnits # DR140
  and DoneUnits = DR320
  or GRADES:DR220>WeightedResults = Unsatisfactory
then
  RECOMMEND = DR140<0.30>.
  NoteScript = true.
endif.

```

DR34 diagnosis:

\ Students who have good results in the Dance unit DR240
 \ and are interested in further Dance
 \ are recommended to DR340.

```

if
  GRADES:DR240>WeightedResults = Satisfactory
  and INTERESTS = Dance
  and DoneUnits # DR340
then
  RECOMMEND = DR340.
endif.

```

DR34_21 diagnosis:

\ Students who are interested in further Dance units
 \ but have either done the highest dance unit DR340
 \ or did not get satisfactory grades in their previous dance unit DR240
 \ are recommended to move across to performance units
 \ with a message to explain this.

```

if
  STAGE = Stage three
  and INTERESTS = Dance
  and DoneUnits # DR210
  and DoneUnits = DR340
  or GRADES:DR240>WeightedResults = Unsatisfactory
then
  RECOMMEND = DR210<0.50>.
  NoteDance = true.
endif.

```

DR34_22 diagnosis:

\ Students who are interested in further Dance units
 \ but have either done the highest dance unit DR340
 \ or did not get satisfactory grades in their previous dance unit DR240
 \ are recommended to move across to performance units
 \ with a message to explain this.

```

if
  STAGE = Stage three
  and INTERESTS = Dance
  and DoneUnits # DR220
  and DoneUnits = DR340
  or GRADES:DR240>WeightedResults = Unsatisfactory
then
  RECOMMEND = DR220<0.50>.
  NoteDance = true.
endif.

```

```

DR34_23 diagnosis:
\ Students who are interested in further Dance units
\ but have either done the highest dance unit DR340
\ or did not get satisfactory grades in their previous dance unit DR240
\ are recommended to move across to performance units
\ with a message to explain this.
    if
    STAGE = Stage three
    and INTERESTS = Dance
    and DoneUnits # DR230
    and DoneUnits = DR340
    or GRADES:DR240>WeightedResults = Unsatisfactory
    then
    RECOMMEND = DR230<0.50>.
    NoteDance = true.
    endif.

\ *****

\      Output messages

NPerformance details:
    if
    NotePerformance = true
    then
    reassert NOTES1 = "*** You wanted to do more Performance".
    reassert NOTES2 = " but you have completed all eligible units".
    reassert NOTES3 = " so we are recommending alternatives. ***".
    endif.
NScript details:
    if
    NoteScript = true
    then
    reassert NOTES1 = "*** You wanted to do more Scriptwriting".
    reassert NOTES2 = " but you have completed all eligible units".
    reassert NOTES3 = " so we are recommending alternatives. ***".
    endif.
NDesign details:
    if
    NoteDesign = true
    then
    reassert NOTES1 = "*** You wanted to do more Design".
    reassert NOTES2 = " but you have completed all eligible units".
    reassert NOTES3 = " so we are recommending alternatives. ***".
    endif.
NDance details:
    if
    NoteDance = true
    then
    reassert NOTES1 = "*** You wanted to do more Dance".
    reassert NOTES2 = " but you have completed all eligible units".
    reassert NOTES3 = " so we are recommending alternatives. ***".
    endif.

\ *****

```

Y8_Compulsory:

```

if    DRAMAJssc lt 3
    and StudYear = 8
then  reassert TIMING = "Compulsory".
endif.

```

Y9_Compulsory:

```

if    DRAMAJssc lt 3
    and StudYear = 9
then  reassert TIMING = "Compulsory".
endif.

```

Y9_Important:

```

if    DRAMAJssc lt 2
    and StudYear = 9
then  reassert TIMING = "Important and Compulsory".
endif.

```

Y9_Urgent:

```

if    DRAMAJssc lt 1
    and StudYear = 9
then  reassert TIMING = "Urgent and Compulsory".
endif.

```

Y10_Important:

```

if    DRAMAJssc lt 3
    and StudYear = 10
then  reassert TIMING = "Important and Compulsory".
endif.

```

Y10_Urgent:

```

if    DRAMAJssc lt 2
    and StudYear = 10
then  reassert TIMING = "Urgent and Compulsory".
endif.

```

jssc_Extension timing:

```

if    DRAMAJssc ge 3
then  reassert TIMING = "Extension".
endif.

```

Drama_Extension timing:

```

if    DRAMAcmplt ge 2
then  reassert TIMING = "Extension".
endif.

```

Specialisation timing:

```

if    DRAMAcmplt ge 4
then  reassert TIMING = "Specialisation".
endif.

```



```
%
```

```
demons:
```

```
Stage3 Demon:
```

```
  when DoneUnits = DR210 | DR220 | DR230 | DR240
  then STAGE = Stage three.
  endwhen.
```

```
All_Drama_Done Demon:
```

```
  when DRAMAcplt = 10
  then RECOMMEND = No_DR.
     NOTES1 = "*** You have completed every Drama unit".
     NOTES2 = " see Mr Dias about Senior Course options.".
     Initial Interests = Performance.
     INTERESTS = Performance.
     FEELINGS = Really keen.
  endwhen.
```

```
\ *****
```

```
%
```

```
actions:
```

```
\ students have to complete three Performing Arts for their jssc
\ at least one from each of Music and Drama
\ so read the Music file for jssc count
```

```
assertclass GRADES = MS110, MS120, MS210, MS220, MS230, MS240, MS250,
MS310, MS311, MS320, MS321, MS330, MS410, No Music, MSCM, MSIL,
MS245.
```

```
read "KES-STU-DATA-MS", DoneUnits.
```

```
  DRAMAJssc = 0.
```

```
  if DoneUnits = MS110
  then reassert DRAMAJssc = DRAMAJssc + 1.
  endif.
  if DoneUnits = MS120 | MS240 | MSIL
  then reassert DRAMAJssc = DRAMAJssc + 1.
  endif.
```

```
erase DoneUnits.
eraseclass GRADES.
```

```
read "KES-STU-DATA-DR", StudSName, StudFName, StudYear, CurrTerm,
GRADES, GRADES(REULTS), DoneUnits.
```

```

\ *****
\      Interpretation of raw data from student CBASS records.

forall G:GRADES do
  while G>RESULTS = N do
    erase G>RESULTS.
    message BANNER.
  message "Our recommendations take into account your previous units and grades".
  message "We already know your past results, but not this term's".
  message "Please enter the grade you reasonably expect for DRAMA this term".
  askfor G>RESULTS.
  endwhile.
  if G>RESULTS = A
    then G>WeightedResults = Satisfactory <1.00>.    endif.
  if G>RESULTS = B
    then G>WeightedResults = Satisfactory <0.85>.    endif.
  if G>RESULTS = C
    then G>WeightedResults = Unsatisfactory <0.10>.  endif.
  if G>RESULTS = D
    then G>WeightedResults = Unsatisfactory <0.25>.  endif.
  if G>RESULTS = E
    then G>WeightedResults = Unsatisfactory <1.00>.  endif.
  if G>RESULTS = W
    then G>WeightedResults = Satisfactory <0.00>.    endif.
endforall.

\ *****
\      Counting JSSC credits and units satisfactorily completed

DRAMAcmplt = 0.

\      if      DoneUnits = DR110 | DR120 | DR130 | DR140
\      then reassert DRAMAjssc = DRAMAjssc + 1.
\      endif.

forall G:GRADES do
  if G>WeightedResults = Satisfactory | Unsatisfactory
  then reassert DRAMAcmplt = DRAMAcmplt + 1.
  endif.
endforall.

if DoneUnits = No Drama
then reassert DRAMAcmplt = DRAMAcmplt - 1.
endif.

if DRAMAcmplt = 1
then reassert DRAMAjssc = DRAMAjssc + 1.
endif.
if DRAMAcmplt = 2
then reassert DRAMAjssc = DRAMAjssc + 2.
endif.

```

```

\ *****
obtain RECOMMEND.

\ *****

\      Screen output.

message BANNER.
message combine ("                Drama advice summary - ", StudFName).
message "Units thus far ", DoneUnits, " ".

message combine ("jssc ",DRAMAjssc," DRAMAcmplt ", DRAMAcmplt).
\message combine ("Interest in Drama - ", FEELINGS).
\if FEELINGS # Prefer not | Definitely not
\      then
\      message combine ("Requested units - ", Initial Interests).      endif.
if NOTES1 # " "
      then
      message NOTES1.
      message NOTES2.
      message NOTES3.
      message NOTES4, " ". endif.

if DRAMAcmplt lt 10
then
message combine ("We recommend that you apply for ", RECOMMEND).
message " ".
message combine ("This recommendation is for ",TIMING," unit/s.").
endif.

\ *****

\      File output.

write "RcmdDR", TIMING, RECOMMEND.

\ *****

\      Termination

if INFO = Justify      then justify RECOMMEND.
                        message HOLD.
                        stop.
                        endif.
if INFO = Again      then nextcase.
                        endif.
if INFO = Continue   then stop.
                        endif.

%
```

\ NEXT-ENGLISH.

constants:

BANNER:

```
"*****",
"
```

HOLD:

```
"           Please press c to continue".
```

%

types:

units:mlt

```
(EN101, EN102, EN103, EN104, EN105, EN111, EN112, EN121, EN122, EN123,
EN131, EN132, EN133, EN134, EN135, EN201, EN202, EN203, EN204, EN205,
EN206, EN207, EN211, EN212, EN213, EN214, EN215, EN221, EN231, EN300,
EN331, EN301, EN302, EN303, EN304, EN305, EN306, EN308, EN314, EN315,
EN317, EN312, EN307, EN309, EN318, EN2P, No English, No_EN).
```

%

attributes:

StudSName:str [default: ""].

StudFName:str [default: ""].

StudYear:int.

CurrTerm:int.

NOTES1:str.

NOTES2:str.

NOTES3:str.

TIMING:str.

StageOne LevelOne:truth.

StageOne LevelTwo:truth.

StageOne LevelThree:truth.

StageTwo LevelOne:truth.

StageTwo LevelTwo:truth.

StageTwo LevelThree:truth.

StageThree LevelOne:truth.

StageThree LevelTwo:truth.

StageThree LevelThree:truth.

\Stage2group:str.

Stage3group1:str.

Stage3group2:str.

Stage3group3:str.

\NotePoetry:truth [default: false].

\NoteAdventureNovels:truth [default: false].

\NoteScienceFiction:truth [default: false].

\NoteRomanceNovels:truth [default: false].

DoneUnits:units.

RECOMMEND:units.

ENGLISHjssc:int [default:0].
 ENGLISHcmplt:int [default: 0].

FEELINGS:sgl
 (Really keen, Enjoy it, Its okay, Prefer not, Definitely not)
 {question: "What are you feelings towards doing English next term?"," "
 {explain: "We only recommend English extension units to those students who wish to
 study more English!"}.

STAGE:sgl
 (Stage one, Stage two, Stage three, Stage four)[default: Stage one].

INFO:sgl
 (Justify
 {question: "Explain the recommendation."},
 Again
 {question: "Enter new information about your preferences."},
 Continue
 {question: "Continue with the next subject area."})
 {question: "What would you like to do now?"," "}

%
 classes:

GRADES:
 attributes:
 RESULTS:sgl (A, B, C, D, E, W, N).
 WeightedResults:sgl (Satisfactory, Poor, Unsatisfactory).
 %
 endclass.

%
 rules:

Setup:
 if true then
 NOTES1 = " "
 NOTES2 = " "
 NOTES3 = " "
 STAGE = Stage one.
 StageOne LevelOne = false.
 \ StageOne LevelTwo = false.
 \ StageOne LevelThree = false.
 StageTwo LevelOne = false.
 StageTwo LevelTwo = false.
 StageTwo LevelThree = false.
 StageThree LevelOne = false.
 StageThree LevelTwo = false.
 StageThree LevelThree = false.
 Stage3group1 = "not completed".
 Stage3group2 = "not completed".
 Stage3group3 = "not completed".
 TIMING = "nil "
 endif.

```

reset flags:
  if STAGE = Stage three
  then reassert StageTwo LevelOne = false.
      reassert StageTwo LevelTwo = false.
      reassert StageTwo LevelThree = false.
  endif.

\ *****

\ Stage 1 status

\ determining Stage 1 - Level 1, Level 2 or Level 3
\ determining advance to Stage 2

S1cL1 diagnosis:
\ If a student achieves an A or B in the two compulsory stage one units
\ then the student has level one status.
\ If the student has only completed these two compulsory units
\ then they are still in stage one.
  if GRADES:EN101>RESULTS = A | B
    and GRADES:EN102>RESULTS = A | B
    and ENGLISHcmplt lt 3
  then
    reassert StageOne LevelOne = true.
    reassert STAGE = Stage one.
  endif.

S1oL1 diagnosis:
\ If a student achieves one of the following grades in optional units
\ then the student has level one status
\ and has completed the stage one requirements
  if GRADES:EN103>RESULTS = A
    or GRADES:EN104>RESULTS = A
    or GRADES:EN105>RESULTS = A
    or GRADES:EN111>RESULTS = A | B | C
    or GRADES:EN112>RESULTS = A | B | C
    or GRADES:EN121>RESULTS = A
    or GRADES:EN122>RESULTS = A
    or GRADES:EN123>RESULTS = A
  then
    reassert StageOne LevelOne = true.
    reassert STAGE = Stage two.
  endif.

S1cL2 diagnosis:
\ If a student achieves a CorD in either of the two compulsory stage one units
\ then the student has level two status.
  if GRADES:EN101>RESULTS = C | D
    or GRADES:EN102>RESULTS = C | D
    and ENGLISHcmplt lt 4
  then
    StageOne LevelTwo = true.
    reassert STAGE = Stage one.
  endif.

```

S1oL2 diagnosis:

\ If a student achieves one of the following grades in optional units

\ then the student has level two status

```

if      GRADES:EN103>RESULTS = B | C
      or GRADES:EN104>RESULTS = B | C
      or GRADES:EN105>RESULTS = B | C
      or GRADES:EN111>RESULTS = D | E
      or GRADES:EN112>RESULTS = D | E
      or GRADES:EN121>RESULTS = B | C
      or GRADES:EN122>RESULTS = B | C
      or GRADES:EN123>RESULTS = B | C
      or GRADES:EN131>RESULTS = A
      or GRADES:EN132>RESULTS = A
      or GRADES:EN133>RESULTS = A
      or GRADES:EN134>RESULTS = A
      or GRADES:EN135>RESULTS = A

```

then

```

StageOne LevelTwo = true.
reassert STAGE = Stage one.
endif.

```

S1_L2_S2 diagnosis:

\ When a Level two student has completed the two compulsory units

\ plus two other stage one units

\ then the student may advance to Stage two.

```

if      StageOne LevelTwo = true
      and ENGLISHcmplt ge 4
then    reassert STAGE = Stage two.
endif.

```

S1cL3 diagnosis:

\ If a student achieves a E in either of the two compulsory stage one units

\ then the student has level three status.

```

if      GRADES:EN101>RESULTS = E
      and GRADES:EN102>RESULTS = E

```

then

```

StageOne LevelThree = true.
reassert STAGE = Stage one.
endif.

```

S1oL3 diagnosis:

\ If a student achieves one of the following grades in optional units

\ then the student has level three status

```

if      GRADES:EN103>RESULTS = D | E
      or GRADES:EN104>RESULTS = D | E
      or GRADES:EN105>RESULTS = D | E
      or GRADES:EN121>RESULTS = D | E
      or GRADES:EN122>RESULTS = D | E
      or GRADES:EN123>RESULTS = D | E
      or GRADES:EN131>RESULTS = B | C | D | E
      or GRADES:EN132>RESULTS = B | C | D | E
      or GRADES:EN133>RESULTS = B | C | D | E
      or GRADES:EN134>RESULTS = B | C | D | E
      or GRADES:EN135>RESULTS = B | C | D | E

```

then

```

StageOne LevelThree = true.
endif.

```

S1_L3_S2 diagnosis:

```
\ When a Level three student has completed the two compulsory units
\ plus three other stage one units
\ then the student may advance to Stage two.
  if      StageOne LevelThree = true
        and ENGLISHcmplt ge 5
  then    reassert STAGE = Stage two.
        endif.
```

* Stage 1 compulsory units

S101 diagnosis:

```
\ All students at Stage one who have not completed EN101
\ are recommended to enrol in this compulsory unit
  if      STAGE = Stage one
        and DoneUnits # EN101
  then
        RECOMMEND = EN101.
        endif.
```

S102 diagnosis:

```
\ All students at Stage one who have not completed EN102
\ are recommended to enrol in this compulsory unit
  if      STAGE = Stage one
        and DoneUnits # EN102
  then
        RECOMMEND = EN102.
        endif.
```

* Stage 1 optional units - at least one to be selected

E111 diagnosis:

```
\ Students in Stage one working at Level one may enrol in this unit
  if      StageOne LevelOne = true
        and STAGE = Stage one
        and DoneUnits # EN111
  then
        RECOMMEND = EN111<1.00>.
        endif.
```

E112 diagnosis:

```
\ Students in Stage one working at Level one may enrol in this unit
  if      StageOne LevelOne = true
        and STAGE = Stage one
        and DoneUnits # EN112
  then
        RECOMMEND = EN112<1.00>.
        endif.
```


E121 diagnosis:

```
\ Students in Stage one working at Level two
\ are recommended to this unit
\ above recommendations for general units
  if      StageOne LevelTwo = true
        and STAGE = Stage one
        and DoneUnits # EN121

  then
    RECOMMEND = EN121<1.00>.
    RECOMMEND = EN103<0.50> & EN104<0.50> & EN105<0.50>.
  endif.
```

E122 diagnosis:

```
\ Students in Stage one working at Level two
\ are recommended to this unit
\ above recommendations for general units
  if      StageOne LevelTwo = true
        and STAGE = Stage one
        and DoneUnits # EN122

  then
    RECOMMEND = EN122<1.00>.
    RECOMMEND = EN103<0.50> & EN104<0.50> & EN105<0.50>.
  endif.
```

E123 diagnosis:

```
\ Students in Stage one working at Level two
\ are recommended to this unit
\ above recommendations for general units
  if      StageOne LevelTwo = true
        and STAGE = Stage one
        and DoneUnits # EN123

  then
    RECOMMEND = EN123<1.00>.
    RECOMMEND = EN103<0.50> & EN104<0.50> & EN105<0.50>.
  endif.
```

E131 diagnosis:

```
\ Students in Stage one working at Level three
\ are recommended to this unit
\ above recommendations for general units
  if      StageOne LevelThree = true
        and STAGE = Stage one
        and DoneUnits # EN131

  then
    RECOMMEND = EN131<1.00>.
    RECOMMEND = EN103<0.50> & EN104<0.50> & EN105<0.50>.
  endif.
```

E132 diagnosis:

```
\ Students in Stage one working at Level three
\ are recommended to this unit
\ above recommendations for general units
  if      StageOne LevelThree = true
        and STAGE = Stage one
        and DoneUnits # EN132

  then
    RECOMMEND = EN132<1.00>.
    RECOMMEND = EN103<0.50> & EN104<0.50> & EN105<0.50>.
  endif.
```

```

E133 diagnosis:
\ Students in Stage one working at Level three
\ are recommended to this unit
\ above recommendations for general units
  if      StageOne LevelThree = true
        and STAGE = Stage one
        and DoneUnits # EN133
  then
    RECOMMEND = EN133.
    RECOMMEND = EN103<0.50> & EN104<0.50> & EN105<0.50>.
  endif.
E134 diagnosis:
\ Students in Stage one working at Level three
\ are recommended to this unit
\ above recommendations for general units
  if      StageOne LevelThree = true
        and STAGE = Stage one
        and DoneUnits # EN134
  then
    RECOMMEND = EN134<1.00>.
    RECOMMEND = EN103<0.50> & EN104<0.50> & EN105<0.50>.
  endif.
E135 diagnosis:
\ Students in Stage one working at Level three
\ are recommended to this unit
\ above recommendations for general units
  if      StageOne LevelThree = true
        and STAGE = Stage one
        and DoneUnits # EN135
  then
    RECOMMEND = EN135<1.00>.
    RECOMMEND = EN103<0.50> & EN104<0.50> & EN105<0.50>.
  endif.

\ *****

\      STAGE TWO

\      determining Level 1, Level 2 or Level 3 status

S2cL1 diagnosis:
\ If a student achieves an A or B in one of the compulsory stage two units
\ then the student has level one status.
  if      GRADES:EN201>RESULTS = A | B
        or GRADES:EN202>RESULTS = A | B
        or GRADES:EN203>RESULTS = A | B
  then
    StageTwo LevelOne = true.
  endif.

```

S2oL1 diagnosis:

```

\ If a student achieves one of the following grades in optional units
\ then the student has level one status
  if    STAGE = Stage two
        and GRADES:EN204>RESULTS = A
        or GRADES:EN205>RESULTS = A
        or GRADES:EN206>RESULTS = A
        or GRADES:EN207>RESULTS = A
        or GRADES:EN211>RESULTS = A | B | C
        or GRADES:EN212>RESULTS = A | B | C
        or GRADES:EN213>RESULTS = A | B | C
        or GRADES:EN214>RESULTS = A | B | C
        or GRADES:EN215>RESULTS = A | B | C
        or GRADES:EN221>RESULTS = A
  then
    StageTwo LevelOne = true.
  endif.

```

S2cL2 diagnosis:

```

\ If a student achieves a C in one of the compulsory stage two units
\ then the student has level two status.
  if    GRADES:EN201>RESULTS = C
        or GRADES:EN202>RESULTS = C
        or GRADES:EN203>RESULTS = C
  then
    reassert StageTwo LevelTwo = true.
  endif.

```

S2oL2 diagnosis:

```

\ If a student achieves one of the following grades in optional units
\ then the student has level two status
  if    STAGE = Stage two
        and GRADES:EN204>RESULTS = B | C
        or GRADES:EN205>RESULTS = B | C
        or GRADES:EN206>RESULTS = B | C
        or GRADES:EN207>RESULTS = B | C
        or GRADES:EN211>RESULTS = C | D | E
        or GRADES:EN212>RESULTS = C | D | E
        or GRADES:EN213>RESULTS = C | D | E
        or GRADES:EN214>RESULTS = C | D | E
        or GRADES:EN215>RESULTS = C | D | E
  then
    reassert StageTwo LevelTwo = true.
  endif.

```

S2cL3 diagnosis:

```

\ If a student achieves a D or E in one of the compulsory stage two units
\ then the student has level three status.
  if    GRADES:EN201>RESULTS = D | E
        or GRADES:EN202>RESULTS = D | E
        or GRADES:EN203>RESULTS = D | E
  then
    reassert StageTwo LevelThree = true.
  endif.

```

S2oL3 diagnosis:

```
\ If a student achieves one of the following grades in optional units
\ then the student has level three status
  if      STAGE = Stage two
         and GRADES:EN204>RESULTS = D | E
         or GRADES:EN205>RESULTS = D | E
         or GRADES:EN206>RESULTS = D | E
         or GRADES:EN207>RESULTS = D | E
         or GRADES:EN221>RESULTS = D | E
         or GRADES:EN231>RESULTS = A | B | C | D | E
  then
    reassert StageTwo LevelThree = true.
  endif.
```

* Stage 2 compulsory unit

S2G diagnosis:

```
\ If a student is at Stage 2
\ and has not done the (one of the three) compulsory unit
\ then recommend they do one of these units
  if      STAGE = Stage two
         and DoneUnits # EN201 | EN202 | EN203
  then    RECOMMEND = EN201 & EN202 & EN203.
  endif.
```

* Stage 2 Level 1 units

E211 diagnosis:

```
\ If a student is at Stage 2 level 1
\ and has not completed this Level 1 unit
\ then recommend this Level 1 unit
\ above recommending general units
  if      StageTwo LevelOne
         and DoneUnits # EN211
         and ENGLISHcmplt lt 7
  then
    RECOMMEND = EN211<1.00>.
\    RECOMMEND = EN204<0.50> & EN205<0.50> & EN206<0.50> &
\    EN207<0.50>.
  endif.
```

E212 diagnosis:

```
\ If a student is at Stage 2 level 1
\ and has not completed this Level 1 unit
\ then recommend this Level 1 unit
\ above recommending general units
  if      StageTwo LevelOne
         and DoneUnits # EN212
         and ENGLISHcmplt lt 7
  then
    RECOMMEND = EN212<1.00>.
\    RECOMMEND = EN204<0.50> & EN205<0.50> & EN206<0.50> &
\    EN207<0.50>.
  endif.
```

E213 diagnosis:

```

\ If a student is at Stage 2 level 1
\ and has not completed this Level 1 unit
\ then recommend this Level 1 unit
\ above recommending general units
    if      StageTwo LevelOne
           and DoneUnits # EN213
           and ENGLISHcmplt lt 7
    then
        RECOMMEND = EN213<1.00>.
\      RECOMMEND = EN204<0.50> & EN205<0.50> & EN206<0.50> &
        EN207<0.50>.
    endif.

```

E214 diagnosis:

```

\ If a student is at Stage 2 level 1
\ and has not completed this Level 1 unit
\ then recommend this Level 1 unit
\ above recommending general units
    if      StageTwo LevelOne
           and DoneUnits # EN214
           and ENGLISHcmplt lt 7
    then
        RECOMMEND = EN214<1.00>.
\      RECOMMEND = EN204<0.50> & EN205<0.50> & EN206<0.50> &
        EN207<0.50>.
    endif.

```

E215 diagnosis:

```

\ If a student is at Stage 2 level 1
\ and has not completed this Level 1 unit
\ then recommend this Level 1 unit
\ above recommending general units
    if      StageTwo LevelOne
           and DoneUnits # EN215
           and ENGLISHcmplt lt 7
    then
        RECOMMEND = EN215<1.00>.
\      RECOMMEND = EN204<0.50> & EN205<0.50> & EN206<0.50> &
        EN207<0.50>.
    endif.

```

E204 diagnosis:

```

\ If a student is at Stage 2 level 2
\ and has not completed this Level 2 unit
\ then recommend this Level 2 unit
    if      StageTwo LevelTwo
           and ENGLISHcmplt lt 7
           and DoneUnits # EN204
    then
        RECOMMEND = EN204<0.50>.
    endif.

```

E205 diagnosis:

```
\ If a student is at Stage 2 level 2
\ and has not completed this Level 2 unit
\ then recommend this Level 2 unit
  if      StageTwo LevelTwo
         and ENGLISHcmplt lt 7
         and DoneUnits # EN205
  then
    RECOMMEND = EN205<0.50>.
  endif.
```

E206 diagnosis:

```
\ If a student is at Stage 2 level 2
\ and has not completed this Level 2 unit
\ then recommend this Level 2 unit
  if      StageTwo LevelTwo
         and ENGLISHcmplt lt 7
         and DoneUnits # EN206
  then
    RECOMMEND = EN206<0.50>.
  endif.
```

E207 diagnosis:

```
\ If a student is at Stage 2 level 2
\ and has not completed this Level 2 unit
\ then recommend this Level 2 unit
  if      StageTwo LevelTwo
         and ENGLISHcmplt lt 7
         and DoneUnits # EN207
  then
    RECOMMEND = EN207<0.50>.
  endif.
```

E221 diagnosis:

```
\ If a student is at Stage 2 level 2
\ and has not completed this Level 2 unit
\ then recommend this Level 2 unit
\ above recommending general units
  if      StageTwo LevelTwo
         and StageOne LevelOne # true
         and ENGLISHcmplt lt 7
         and DoneUnits # EN221
  then
    RECOMMEND = EN221<1.00>.
  endif.
```

E231 diagnosis:

```
\ If a student is at Stage 2 level 3
\ and has not completed this Level 3 unit
\ then recommend this Level 3 unit
\ above recommending general units
  if      StageTwo LevelThree
         and StageOne LevelOne # true
         and StageOne LevelTwo # true
         and ENGLISHcmplt lt 7
         and DoneUnits # EN231
  then
    RECOMMEND = EN231<1.00>.
    RECOMMEND = EN204<0.50> & EN205<0.50>
                & EN206<0.50> & EN207<0.50>.
  endif.
```

```

E231_2O4 diagnosis:
\ If a student is at Stage 2 level 3
\ and has completed EN231
\ then recommend general units
    if      StageTwo LevelThree
           and DoneUnits = EN231
           and ENGLISHcmplt lt 7
           and DoneUnits # EN204

    then
    RECOMMEND = EN204.
    endif.

```

```

E231_2O5 diagnosis:
\ If a student is at Stage 2 level 3
\ and has completed EN231
\ then recommend general units
    if      StageTwo LevelThree
           and DoneUnits = EN231
           and ENGLISHcmplt lt 7
           and DoneUnits # EN205

    then
    RECOMMEND = EN205.
    endif.

```

```

E231_2O6 diagnosis:
\ If a student is at Stage 2 level 3
\ and has completed EN231
\ then recommend general units
    if      StageTwo LevelThree
           and DoneUnits = EN231
           and ENGLISHcmplt lt 7
           and DoneUnits # EN206

    then
    RECOMMEND = EN206.
    endif.

```

```

E231_2O7 diagnosis:
\ If a student is at Stage 2 level 3
\ and has completed EN231
\ then recommend general units
    if      StageTwo LevelThree
           and DoneUnits = EN231
           and ENGLISHcmplt lt 7
           and DoneUnits # EN207

    then
    RECOMMEND = EN207.
    endif.

```

```

\ *****

```

\ STAGE 3

Stage3 diagnosis:

```

if
    DoneUnits = EN201 | EN202 | EN203
    and ENGLISHcmplt ge 7
then
    reassert STAGE = Stage three.
\    reassert StageTwo LevelOne = false.
\    reassert StageTwo LevelTwo = true.
\    reassert StageTwo LevelThree = false.
endif.

```

* determining level one or level three status

S3_L1 diagnosis:

\ If a student achieves one of the following grades

\ then the student has level one status

```

if
    GRADES:EN301>RESULTS = A | B
    or GRADES:EN302>RESULTS = A | B
    or GRADES:EN303>RESULTS = A | B
    or GRADES:EN304>RESULTS = A | B
    or GRADES:EN305>RESULTS = A | B
    or GRADES:EN306>RESULTS = A | B
    or GRADES:EN307>RESULTS = A | B
    or GRADES:EN309>RESULTS = A | B
    or GRADES:EN314>RESULTS = A | B | C
    or GRADES:EN315>RESULTS = A | B | C
    or GRADES:EN317>RESULTS = A | B | C
    or GRADES:EN312>RESULTS = A | B | C
    or GRADES:EN318>RESULTS = A | B | C

```

then

```

reassert StageThree LevelOne = true.
reassert StageThree LevelTwo = false.
endif.

```

S3_L3 diagnosis:

\ If a student achieves one of the following grades

\ then the student has level three status

```

if
    GRADES:EN301>RESULTS = D | E
    or GRADES:EN302>RESULTS = D | E
    or GRADES:EN303>RESULTS = D | E
    or GRADES:EN304>RESULTS = D | E
    or GRADES:EN305>RESULTS = D | E
    or GRADES:EN306>RESULTS = D | E
    or GRADES:EN307>RESULTS = D | E
    or GRADES:EN309>RESULTS = D | E

```

then

```

reassert StageThree LevelThree = true.
reassert StageThree LevelTwo = false.
endif.

```


* optional group

E300 diagnosis:

```
\ If a student has completed the jssc requirements
\ and has not done this unit
\ and has achieved level one status
\ then recommend this optional unit
  if
    StudYear = 10
    and ENGLISHjssc = 9
    and DoneUnits # EN300
    and StageThree LevelThree = false
  then
    RECOMMEND = EN300.
  endif.
```

E331 diagnosis:

```
\ If a student has completed the jssc requirements
\ and has not done this unit
\ and has achieved level three status
\ then recommend this optional unit
  if
    ENGLISHjssc = 9
    and DoneUnits # EN331
    and StageThree LevelThree = true
  then
    RECOMMEND = EN331.
  endif.
```

* group one - one compulsory others optional

S3_G1 compulsory diagnosis:

```
\ If a student has completed Stage Two
\ but has not completed a unit from stage three group one
\ then recommend one of these units
  if
    STAGE = Stage three
    and Stage3group1 = "not completed"
  then
    RECOMMEND = EN301 & EN302 & EN303.      endif.
```

E301 diagnosis:

```
\ If a student has completed the jssc requirements
\ and has not done this unit
\ then recommend this optional unit
  if
    ENGLISHjssc = 9
    and DoneUnits # EN301
  then
    RECOMMEND = EN301.      endif.
```

E302 diagnosis:

```
\ If a student has completed the jssc requirements
\ and has not done this unit
\ then recommend this optional unit
  if
    ENGLISHjssc = 9
    and DoneUnits # EN302
  then
    RECOMMEND = EN302.      endif.
```

E303 diagnosis:

```
\ If a student has completed the jssc requirements
\ and has not done this unit
\ then recommend this optional unit
  if
    ENGLISHjssc = 9
    and DoneUnits # EN303
  then
    RECOMMEND = EN303.
  endif.
```

* group two - one compulsory others optional

S3_G2_L0 compulsory diagnosis:

```
\ If a student has completed Stage two
\ but has not completed a unit from Stage three Group two
\ and the student is not Level one
\ then recommend one of these units
  if
    STAGE = Stage three
    and Stage3group2 = "not completed"
    and StageThree LevelOne # true
  then
    RECOMMEND = EN304 & EN305 & EN306 & EN308.
  endif.
```

S3_G2_L1 compulsory diagnosis:

```
\ If a student has completed Stage two
\ but has not completed a unit from Stage three Group two
\ and the student is Level one
\ then recommend one of these units
  if
    STAGE = Stage three
    and Stage3group2 # "completed"
    and StageThree LevelOne = true
  then
    RECOMMEND = EN314 & EN315 & EN317 & EN312.
  endif.
```

E304 diagnosis:

```
\ If a student has completed the jssc requirements
\ and has not done this unit
\ and has not achieved level one status
\ then recommend this optional unit
  if
    ENGLISHjssc = 9
    and DoneUnits # EN304
    and StageThree LevelOne # true
  then
    RECOMMEND = EN304.
  endif.
```

```

E305 diagnosis:
\ If a student has completed the jssc requirements
\ and has not done this unit
\ and has not achieved level one status
\ then recommend this optional unit
  if
    ENGLISHjssc = 9
    and DoneUnits # EN305
    and StageThree LevelOne # true
  then
    RECOMMEND = EN305.
  endif.

```

```

E306 diagnosis:
\ If a student has completed the jssc requirements
\ and has not done this unit
\ and has not achieved level one status
\ then recommend this optional unit
  if
    ENGLISHjssc = 9
    and DoneUnits # EN306
    and StageThree LevelOne # true
  then
    RECOMMEND = EN306.
  endif.

```

```

E308 diagnosis:
\ If a student has completed the jssc requirements
\ and has not done this unit
\ and has not achieved level one status
\ then recommend this optional unit
  if
    ENGLISHjssc = 9
    and DoneUnits # EN308
    and StageThree LevelOne # true
  then
    RECOMMEND = EN308.
  endif.

```

```

E314 diagnosis:
\ If a student has completed the jssc requirements
\ and has not done this unit
\ and has achieved level one status
\ then recommend this optional unit
  if
    ENGLISHjssc = 9
    and DoneUnits # EN314
    and StageThree LevelOne = true
  then
    RECOMMEND = EN314.
  endif.

```

```

E315 diagnosis:
\ If a student has completed the jssc requirements
\ and has not done this unit
\ and has achieved level one status
\ then recommend this optional unit
  if
    ENGLISHjssc = 9
    and DoneUnits # EN315
    and StageThree LevelOne = true
  then
    RECOMMEND = EN315.
  endif.

```

```

E317 diagnosis:
\ If a student has completed the jssc requirements
\ and has not done this unit
\ and has achieved level one status
\ then recommend this optional unit
  if
    ENGLISHjssc = 9
    and DoneUnits # EN317
    and StageThree LevelOne = true
  then
    RECOMMEND = EN317.
  endif.

```

```

E312 diagnosis:
\ If a student has completed the jssc requirements
\ and has not done this unit
\ and has achieved level one status
\ then recommend this optional unit
  if
    ENGLISHjssc = 9
    and DoneUnits # EN312
    and StageThree LevelOne = true
  then
    RECOMMEND = EN312.
  endif.

```

* group three - one compulsory others optional

```

S3_G3_L0 compulsory diagnosis:
\ If a student has completed Stage Two
\ but has not completed a unit from Stage three Group three
\ and the student is not Level one
\ then recommend one of these units
  if
    STAGE = Stage three
    and Stage3group3 = "not completed"
    and StageThree LevelOne = false
  then
    RECOMMEND = EN307 & EN309.
  endif.

```

```

S3_G3_L1 compulsory diagnosis:
\ If a student has completed Stage Two
\ but has not completed a unit from Stage three Group three
\ and the student is working Level one
\ then recommend one of these units
    if
        STAGE = Stage three
        and Stage3group3 # "completed"
        and StageThree LevelOne = true
    then
        RECOMMEND = EN318.
    endif.

```

```

E307 diagnosis:
\ If a student has completed the jssc requirements
\ and has not done this unit
\ and has not achieved level one status
\ then recommend this optional unit
    if
        ENGLISHjssc = 9
        and DoneUnits # EN307
        and StageThree LevelOne # true
    then
        RECOMMEND = EN307.
    endif.

```

```

E309 diagnosis:
\ If a student has completed the jssc requirements
\ and has not done this unit
\ and has not achieved level one status
\ then recommend this optional unit
    if
        ENGLISHjssc = 9
        and DoneUnits # EN309
        and StageThree LevelOne # true
    then
        RECOMMEND = EN309.
    endif.

```

```

E318 diagnosis:
\ If a student has completed the jssc requirements
\ and has not done this unit
\ and has achieved level one status
\ then recommend this optional unit
    if
        ENGLISHjssc = 9
        and DoneUnits # EN318
        and StageThree LevelOne = true
    then
        RECOMMEND = EN318.
    endif.

```

```

\ *****

```

\ Output messages

Important diagnosis:

```
if
  STAGE = Stage two
then
  reassert TIMING = "Important and Compulsory".
endif.
```

UrgentPriority diagnosis:

```
if
  STAGE = Stage one
  or RECOMMEND = EN300
then
  reassert TIMING = "Urgent and Compulsory".
endif.
```

Sjssc9 diagnosis:

```
if
  ENGLISHjssc lt 9
then
  reassert TIMING = "Urgent and Compulsory".
endif.
```

Extension diagnosis:

```
if
  ENGLISHjssc = 9
then
  reassert TIMING = "Extension".
endif.
```

Specialisation diagnosis:

```
if
  ENGLISHcmplt ge 12
then
  reassert TIMING = "Specialisation".
endif.
```

Year10_Exam diagnosis:

```
if
  StudYear = 10
  and CurrTerm = 3
then
  reassert TIMING = "Urgent and Compulsory".
  reassert NOTES1 = "*** Year Ten students must do English in the exam term !".
endif.
```

\ *****

%
demonstrations:

\ flag to ensure a unit from each Stage 3 compulsory group is completed
\ before other Stage 3 units are recommended

```

S3_G1 Demon:
  when DoneUnits = EN301 | EN302 | EN303
  then Stage3group1 = "completed".
  endwhen.

S3_G2 Demon:
  when DoneUnits = EN304 | EN305 | EN306 | EN308 | EN314 | EN315 |
  EN317 | EN312
  then Stage3group2 = "completed".
  endwhen.

S3_G3 Demon:
  when DoneUnits = EN307 | EN309 | EN318
  then Stage3group3 = "completed".
  endwhen.

Ejssc9 Demon:
  when
  DoneUnits = EN101 & EN102
  and DoneUnits = EN201 | EN202 | EN203
  and DoneUnits = EN301 | EN302 | EN303
  and DoneUnits = EN304 | EN305 | EN306 | EN308 | EN314 | EN315 | EN317 | EN312
  and DoneUnits = EN307 | EN309 | EN318
  and ENGLISHcmplt ge 9
  then
  ENGLISHjssc = 9.
  endwhen.

No_English Demon:
\ When a student has completed the jssc requirements
\ and next term is not the term for year ten exams
\ and the student does not wish to continue with English
\ then recommend No English
  when
    ENGLISHjssc = 9
    and StudYear # 10
    and CurrTerm # 3
    and FEELINGS = Prefer not | Definitely not
  then
    reassert RECOMMEND = No_EN <0.50>.
  endwhen.

```

```
\ *****
```

```
%
actions:
```

```
read "KES-STU-DATA-EN", StudSName, StudFName, StudYear, CurrTerm,
GRADES, GRADES(REULTS), DoneUnits.
```

```
\ *****
```

\ Interpretation of raw data from student CBASS records.

```
forall G:GRADES do
  while G>RESULTS = N do
    erase G>RESULTS.
    message BANNER.
  message "Our recommendations take into account your previous units and grades".
  message "We already know your past results, but not this term's".
  message "Please enter the grade you reasonably expect for ENGLISH this term".
  askfor G>RESULTS.
  endwhile.
  if G>RESULTS = A
    then G>WeightedResults = Satisfactory <1.00>.    endif.
  if G>RESULTS = B
    then G>WeightedResults = Satisfactory <0.85>.    endif.
  if G>RESULTS = C
    then G>WeightedResults = Satisfactory <0.30>.    endif.
  if G>RESULTS = D
    then G>WeightedResults = Satisfactory <0.10>.    endif.
  if G>RESULTS = E
    then G>WeightedResults = Unsatisfactory <0.50>.  endif.
  if G>RESULTS = W
    then G>WeightedResults = Satisfactory <0.00>.    endif.
endforall.
```

\ Counting units satisfactorily completed

ENGLISHcmplt = 0.

```
forall G:GRADES do
  if G>WeightedResults = Satisfactory | Unsatisfactory
  then reassert ENGLISHcmplt = ENGLISHcmplt + 1.
  endif.
endforall.
```

```
if DoneUnits = No_EN
then reassert ENGLISHcmplt = ENGLISHcmplt - 1.
endif.
```

\ *****

obtain RECOMMEND.

\ *****

\ Screen output.

```
message BANNER.
message combine ("          English recommendation summary - ", StudFName).
message "Units thus far ", DoneUnits, " ".
```

```
message combine ("We recommend that you consider ", RECOMMEND).
message combine ("This recommendation considers ",TIMING," unit/s.").
```

```
if NOTES1 # " "
then  message NOTES1.
      message NOTES2.
      message NOTES3, " ". endif.
```

\ *****

\ File output.

```
write "RcmdEN", TIMING, RECOMMEND.
```

\ *****

\ Termination

```
if INFO = Justify    then justify RECOMMEND.
                    message HOLD.
                    stop.
                    endif.
if INFO = Again      then nextcase.
                    endif.
if INFO = Continue   then stop.
                    endif.
```

%

\ NEXT-HOME ECONOMICS.

constants:

BANNER:

```

"*****",
" ".

```

HOLD:

```

"           Please press c to continue".

```

%

types:

units:mlt

```

(HE110, HE210, HE310, HE311, HE312, HE412, HE120, HE220, HE223, HE321,
HE323, HE421, HE423, No HomeEc, No_HE).

```

%

attributes:

StudSName:str [default: " "].

StudFName:str [default: " "].

StudYear:int.

CurrTerm:int.

NOTES1:str.

NOTES2:str.

NOTES3:str.

TIMING:str.

NoteFood:truth.

NoteGarment:truth.

NoteTextile:truth.

DoneUnits:units.

RECOMMEND:units.

HECjssc:int [default:0].

HECcmplt:int [default: 0].

Initial Interests:sgl

(Food

```

    {question:" Food and Nutrition"}

```

```

    {explain: "These units involve cooking."},

```

Garment

```

    {question:" Designing and Sewing"}

```

```

    {explain: "These units involve making clothes for yourself."},

```

Textile

```

    {question:" Textile crafts"}

```

```

    {explain: "These units involve making non-clothing items."})

```

```

{question: "Which of these HomeEc groups would you prefer next term?"," " }

```

```

{explain: "Our recommendations take in account your preferences as well as your
grades in the pre-requisite units."}.

```

INTERESTS:sgl
(Food, Garment, Textile).

FEELINGS:sgl
(Really keen, Enjoy it, Its okay, Prefer not, Definitely not)
{question: "What are you feelings towards doing HomeEc next term?"," "}
{explain: "We only recommend HomeEc extension units to those students who wish to study more HomeEc!"}.

STAGE:sgl
(Stage one, Stage two, Stage three, Stage four) [default: Stage one].

INFO:sgl
(Justify
 {question: "Explain the recommendation."},
Again
 {question: "Enter new information about your preferences."},
Continue
 {question: "Continue with the next subject area."})
{question: "What would you like to do now?"," "}

%
classes:

GRADES:
attributes:
RESULTS:sgl (A, B, C, D, E, W, N).
WeightedResults:sgl (Satisfactory, Unsatisfactory).
%
endclass.

%
rules:

Setup textoutput:
 if true
 then NOTES1 = " ".
 NOTES2 = " ".
 NOTES3 = " ".
 TIMING = " ".
 endif.

\ *****

\ Stage One compulsory units

H1double diagnosis:
\ If a student has not done HE110 or HE120
\ then recommend both compulsory units
 if
 DoneUnits # HE110 | HE120
 then
 RECOMMEND = HE110 & HE120.
 endif.

H110 diagnosis:

```
\ If a student has done one compulsory unit
\ then recommend the other compulsory unit
  if
    DoneUnits = HE120
    and DoneUnits # HE110
  then
    RECOMMEND = HE110.
  endif.
```

H120 diagnosis:

```
\ If a student has done one compulsory unit
\ then recommend the other compulsory unit
  if
    DoneUnits = HE110
    and DoneUnits # HE120
  then
    RECOMMEND = HE120.
  endif.
```

\ *****

\ Student feelings towards and interests in non-compulsory units

HomeEc_PreferNot response:

```
  if
    STAGE = Stage two | Stage three
    and FEELINGS = Prefer not
  then
    RECOMMEND = No_HE <0.50>.
  endif.
```

HomeEc_DefinitelyNot response:

```
  if
    STAGE = Stage two
    and FEELINGS = Definitely not
  then
    RECOMMEND = No_HE <1.00>.
  endif.
```

Food_keen response:

```
  if
    STAGE = Stage two | Stage three
    and FEELINGS = Really keen
    and Initial Interests = Food
  then
    INTERESTS = Food <1.00>.
  endif.
```

Food_Enjoy response:

```
  if
    STAGE = Stage two | Stage three
    and FEELINGS = Enjoy it
    and Initial Interests = Food
  then
    INTERESTS = Food <0.75>.
  endif.
```

Food_OK response:

```

if
    STAGE = Stage two | Stage three
    and FEELINGS = Its okay
    and Initial Interests = Food
then
    INTERESTS = Food <0.30>.
endif.

```

Garment_keen response:

```

if
    STAGE = Stage two | Stage three
    and FEELINGS = Really keen
    and Initial Interests = Garment
then
    INTERESTS = Garment <1.00>.
endif.

```

Garment_Enjoy response:

```

if
    STAGE = Stage two | Stage three
    and FEELINGS = Enjoy it
    and Initial Interests = Garment
then
    INTERESTS = Garment <0.75>.
endif.

```

Garment_OK response:

```

if
    STAGE = Stage two | Stage three
    and FEELINGS = Its okay
    and Initial Interests = Garment
then
    INTERESTS = Garment <0.30>.
endif.

```

Textile_keen response:

```

if
    STAGE = Stage two | Stage three
    and FEELINGS = Really keen
    and Initial Interests = Textile
then
    INTERESTS = Textile <1.00>.
endif.

```

Textile_Enjoy response:

```

if
    STAGE = Stage two | Stage three
    and FEELINGS = Enjoy it
    and Initial Interests = Textile
then
    INTERESTS = Textile <0.75>.
endif.

```

Textile_OK response:

```

if
    STAGE = Stage two | Stage three
    and FEELINGS = Its okay
    and Initial Interests = Textile
then
    INTERESTS = Textile <0.30>.
endif.

```

\ *****

\ Food extensions

H210 diagnosis:

```
\ Students who have completed both the compulsory units
\ and are interested in further Food studies
\ but have not yet done HE210
\ are recommended to HE210
  if
    DoneUnits = HE110 & HE120
    and INTERESTS = Food
    and DoneUnits # HE210
  then
    RECOMMEND = HE210.
  endif.
```

H310 diagnosis:

```
\ Students who have done well in HE210
\ and are interested in further Food studies
\ are recommended to the Stage 3 units
  if
    GRADES:HE210>WeightedResults = Satisfactory
    and INTERESTS = Food
    and DoneUnits # HE310
  then
    RECOMMEND = HE310.
  endif.
```

H311 diagnosis:

```
\ Students who have done well in HE210
\ and are interested in further Food studies
\ are recommended to the Stage 3 units
  if
    GRADES:HE210>WeightedResults = Satisfactory
    and INTERESTS = Food
    and DoneUnits # HE311
  then
    RECOMMEND = HE311.
  endif.
```

H312 diagnosis:

```
\ Students who have done well in HE210
\ and are interested in further Food studies
\ are recommended to the Stage 3 units
  if
    GRADES:HE210>WeightedResults = Satisfactory
    and INTERESTS = Food
    and DoneUnits # HE312
  then
    RECOMMEND = HE312.
  endif.
```

H412 diagnosis:

```
\ Students who have done well in at least one of the Stage 3 units
\ and are interested in further Food studies
\ are recommended to the Stage 4 unit
  if
    DoneUnits = HE310 | HE311 | HE312
    and DoneUnits # HE412
    and INTERESTS = Food
    and GRADES:HE310>WeightedResults = Satisfactory
    or GRADES:HE311>WeightedResults = Satisfactory
    or GRADES:HE312>WeightedResults = Satisfactory
  then
    RECOMMEND = HE412<1.00>.
  endif.
```

Foodend diagnosis:

```
\ Students who have completed the final Food unit
\ and are interested in further Food studies
\ are not recommended for Home Economics
  if
    DoneUnits = HE412
    and INTERESTS = Food
  then
    RECOMMEND = No_HE.
    NoteFood = true.
  endif.
```

\ *****

\ Garment

H220 diagnosis:

```
\ Students who have completed both the compulsory units
\ and are interested in further Garment studies
\ but have not yet done HE220
\ are recommended to HE220
  if
    DoneUnits = HE110 & HE120
    and INTERESTS = Garment
    and DoneUnits # HE220
  then
    RECOMMEND = HE220.
  endif.
```

H321 diagnosis:

```
\ Students who have done well in HE221
\ and are interested in further Garment studies
\ are recommended to the Stage 3 unit
  if
    GRADES:HE220>WeightedResults = Satisfactory
    and INTERESTS = Garment
    and DoneUnits # HE321
  then
    RECOMMEND = HE321.
  endif.
```

H421 diagnosis:

```
\ Students who have done well in HE321
\ and are interested in further Garment studies
\ are recommended to the Stage 4 unit
  if
    GRADES:HE321>WeightedResults = Satisfactory
    and INTERESTS = Garment
    and DoneUnits # HE421
  then
    RECOMMEND = HE421.
  endif.
```

Garmentend diagnosis:

```
\ Students who have completed the final Garment unit
\ and are interested in further Garment studies
\ are not recommended for Home Economics
  if
    DoneUnits = HE421
    and INTERESTS = Garment
  then
    RECOMMEND = No_HE.
    NoteGarment = true.
  endif.
```

\ *****

\ Textile

H223 diagnosis:

```
\ Students who have completed both the compulsory units
\ and are interested in further Textile studies
\ but have not yet done HE223
\ are recommended to HE223
  if
    DoneUnits = HE110 & HE120
    and INTERESTS = Textile
    and DoneUnits # HE223
  then
    RECOMMEND = HE223.
  endif.
```

H323 diagnosis:

```
\ Students who have done well in HE223
\ and are interested in further Textile studies
\ are recommended to the Stage 3 unit
  if
    GRADES:HE223>WeightedResults = Satisfactory
    and INTERESTS = Textile
    and DoneUnits # HE323
  then
    RECOMMEND = HE323.
  endif.
```


H423 diagnosis:

```
\ Students who have done well in HE323
\ and are interested in further Textile studies
\ are recommended to the Stage 4 unit
  if
    GRADES:HE323>WeightedResults = Satisfactory
    and INTERESTS = Textile
    and DoneUnits # HE423
  then
    RECOMMEND = HE423.
  endif.
```

Textileend diagnosis:

```
\ Students who have completed the final Textile unit
\ and are interested in further Textile studies
\ are not recommended for Home Economics
  if
    DoneUnits = HE423
    and INTERESTS = Textile
  then
    RECOMMEND = No_HE.
    NoteTextile = true.
  endif.
```

\ *****

\ Output messages

NFood details:

```
  if
    NoteFood = true
  then
    reassert NOTES1 = "*** You wanted to do more Food and Nutrition".
    reassert NOTES2 = " but you have gone through to Stage four!".
    reassert NOTES3 = " You may consider other Home Ec topics. ***".
  endif.
```

NGarment details:

```
  if
    NoteGarment = true
  then
    reassert NOTES1 = "*** You wanted to do more Fashion and Design".
    reassert NOTES2 = " but you have completed all appropriate units".
    reassert NOTES3 = " You may consider other Home Ec topics.. ***".
  endif.
```

NTextile details:

```
  if
    NoteTextile = true
  then
    reassert NOTES1 = "*** You wanted to do more Textile Crafts".
    reassert NOTES2 = " but you have completed all appropriate units".
    reassert NOTES3 = " You may consider other Home Ec topics.. ***".
  endif.
```

First_8 diagnosis:

```

if
  HECjssc = 0
  and StudYear = 8
then
  reassert TIMING = "Compulsory".
endif.

```

First_9 diagnosis:

```

if
  HECjssc = 0
  and StudYear = 9
then
  reassert TIMING = "Important and Compulsory".
endif.

```

First_10 diagnosis:

```

if
  HECjssc = 0
  and StudYear = 10
then
  reassert TIMING = "Urgent and Compulsory".
endif.

```

Second_8 diagnosis:

```

if
  HECjssc = 1
  and StudYear = 8
then
  reassert TIMING = "Compulsory".
endif.

```

Second_9 diagnosis:

```

if
  HECjssc = 1
  and StudYear = 9
then
  reassert TIMING = "Compulsory".
endif.

```

Second_10 diagnosis:

```

if
  HECjssc = 1
  and StudYear = 10
then
  reassert TIMING = "Important and Compulsory".
endif.

```

Extension diagnosis:

```

if
  DoneUnits = HE110 & HE120
  and HECcmplt lt 5
then
  reassert TIMING = "Extension".
endif.

```

Specialisation diagnosis:

```

if
  HECcmplt ge 5
then
  reassert TIMING = "Specialisation".
endif.

```

```

%
demons:

Stage2 Demon:
  when
    DoneUnits = HE110 & HE120
  then
    STAGE = Stage two.
  endwhen.
Stage3 Demon:
  when
    DoneUnits = HE110 & HE120
    and DoneUnits = HE210 | HE220 | HE223
  then
    reassert STAGE = Stage three.
  endwhen.

%
actions:

read "KES-STU-DATA-HE", StudSName, StudFName, StudYear, CurrTerm,
GRADES, GRADES(RESULTS), DoneUnits.

\ *****

\ Interpretation of raw data from student CBASS records.

forall G:GRADES do
  while G>RESULTS = N do
    erase G>RESULTS.
    message BANNER.
  message "Our recommendations take into account your previous units and grades".
  message "We already know your past results, but not this term's".
  message "Please enter the grade you reasonably expect for HEC this term".
  askfor G>RESULTS.
  endwhile.
  if G>RESULTS = A
    then G>WeightedResults = Satisfactory <1.00>.   endif.
  if G>RESULTS = B
    then G>WeightedResults = Satisfactory <0.85>.   endif.
  if G>RESULTS = C
    then G>WeightedResults = Satisfactory <0.70>.   endif.
  if G>RESULTS = D
    then G>WeightedResults = Satisfactory <0.50>.   endif.
  if G>RESULTS = E
    then G>WeightedResults = Unsatisfactory <0.50>. endif.
  if G>RESULTS = W
    then G>WeightedResults = Satisfactory <0.10>.   endif.
endforall.

\ *****

```

```

\      Counting JSSC credits and units satisfactorily completed

      HECjssc = 0.
      HECcmplt = 0.

      if      DoneUnits = HE110
      then reassert HECjssc = HECjssc + 1.
      endif.

      if      DoneUnits = HE120
      then reassert HECjssc = HECjssc + 1.
      endif.

      forall G:GRADES do
      if G>WeightedResults = Satisfactory
      then reassert HECcmplt = HECcmplt + 1.
      endif.
      endforall.

      if DoneUnits = No HomeEc
      then reassert HECcmplt = HECcmplt - 1.
      endif.

\ *****

obtain RECOMMEND.

\ *****

\      Screen output.

message BANNER.
message combine (" Home Economics recommendation summary - ", StudFName).
message "Units thus far ", DoneUnits, " ".

if NOTES1 # " "
then  message NOTES1.
      message NOTES2.
      message NOTES3, " ". endif.

message combine ("We recommend that you apply for ", RECOMMEND).
message " ".
message combine ("This recommendation is for ",TIMING," unit/s.").

\ *****

\      File output.

write "RcmdHE", TIMING, RECOMMEND.

\ *****

\      Termination
if INFO = Justify      then justify RECOMMEND.
                        message HOLD.
                        stop.
                        endif.
if INFO = Again       then nextcase.
                        endif.

```