

Information System Based Implementation of Semester-to-Semester Progression at University of Botswana

J.S.J. Daka
University of Botswana, Botswana

ABSTRACT

In August 2002, the University of Botswana changed from a yearlong to a semester based academic system. However, the progression regulations were not be fully implemented on the information system mainly due to problems of interpretation. Due to this, a manual system was used. The manual system could not cope in making recommendations for all students leading to some regulations not being applied. To address this, a revision of the regulations was undertaken in 2007. These regulations require implementation on the information system.

This paper discusses the main aspects of the 2007 regulations and their implementation via a computer program. The paper shows that a complete implementation of the regulations is possible using a computer program. The program was applied to sample students' academic results to provide progression recommendations that were found to agree with expected results. The paper concludes that an information system based implementation ensures uniform application of regulations.

Keywords: *semester-to-semester progression; academic regulations; academic warning; academic probation; fail and discontinue; fail and exclude; information systems.*

INTRODUCTION

In August 2002, the University of Botswana introduced a semester based academic system having been on a yearlong academic system. In a semester system, students are examined at the end of each semester (about 4 months), with two semesters forming an academic year. In a yearlong system, students were examined at the end of an academic year (about nine months). The semester system academic year is structured as semester one, an intervening 3 weeks holiday including the Christmas/New Year's holiday, followed by semester two and a long vacation (about 2 months). The semester system requires that academic results are presented at the end of each semester. In a yearlong system, academic results were presented once at the end of the academic year.

The change to semesterised programmes was coupled to a complete change in regulations to suit the semester system (University of Botswana 2006). Under these regulations, a student proceeded to the next semester if they had passed all core courses and had a cumulative grade point average (CGPA) of at least 2.0. Such a student was put on a proceed (P) status. A student could also proceed with a status of probation (PP) if they had failed a core, pre-requisite or co-requisite with a mark of at least 40 percent and had a CGPA of at least 1.5. A student was put on a fail and discontinue (FD) status if they had failed all courses, had a CGPA of less than 1.5, or

had acquired three probations. These regulations did not specify the fate of students who had a CGPA of at least 1.5 and failed a core, pre-requisite or co-requisite with a mark of less than 40 percent. In addition, the fate of students who had more than one FD status was not specified. A flowchart of the above regulations is shown in Figure A1 in the Appendix section. Note that these regulations did not have a fail and exclude status, this meant that a student could technically continue studying by serving an FD and re-enrolling as many times as required to complete a programme.

The regulations required to be implemented on the existing students' information system. However, the difficulty of interpreting the regulations led to them not being implemented fully. The supplier of the information system adjudged the regulations difficult to understand. This partial implementation led to a manual system being used to handle semester-to-semester progression. The manual system involved printing of student's CGPA obtained from the information system. The list was used to identify students who had a CGPA of less than 1.5 who would then be put on an FD status. The information system also provided a listing of course results for each student which was used to identify students who had failed all courses in a semester who would be assigned an FD status. However, identifying students who were on three probations for an FD status proved difficult. The information for this status required manually checking results of at least three previous semesters for each student. This regulation could not be applied effectively in Faculties leading to its waiver. In addition, the other regulation parts that could be applied were not applied uniformly in Faculties due to difficulties associated with manual checking of the results.

The problems arising from the partial implementation of the regulations resulted in a number of students who should have been assigned the FD status proceeding from one semester to another without any control in most Faculties. This led to a large number of students with an FD status remaining in the university. To rectify the situation, the university revised the 2002 regulations in 2007 (University of Botswana 2007) with assurance that they were to be fully implemented on the students' information system. However, the implementation of these regulations on the information system has not yet been achieved.

The university recognised the need and importance for the implementation of its progression regulations on an information system and put in place business processes to address the issue. Meanwhile, Faculties continue to make students' academic recommendations using the manual system. To rectify this situation, the Faculty of Engineering and Technology embarked on a project to implement the progression regulations using a computer program. This approach would allow the Faculty to process results and make recommendations in a timely manner.

This paper focuses on the design and testing of a computer program for use on the students' information system to implement the revised progression regulations. The use of the system avoids the problems of non-uniform application or non-application of progression regulations as occurred from August 2002 to December 2007.

REVISED PROGRESSION REGULATIONS

To address some of the problems of the 2002 regulations, the university embarked on a review exercise that resulted in the revised semester-to-semester progression regulations (University of Botswana 2007). Under these regulations, a student can proceed to the next semester on academic good standing, on academic warning, and on academic probation. All proceeding students must have passed at least half the number of attempted semester credits. In addition, a student proceeding on good academic standing must have attained a CGPA of at least 2.00. For a student proceeding on academic warning or academic probation, the required CGPA is dependent on the total number of attempted credits as indicated in Table 1. A student proceeding on good academic standing is limited to register for up to 18 credits if such a student carried any failed core, pre/co requisite course(s), otherwise 21 credits with permission from the relevant Dean of Faculty. The credit limit for a student on academic warning or probation is 16 or 14 respectively.

Table 1: Academic Warning/Probation CGPA Threshold Values (University of Botswana 2007)

Attempted Credits	Academic Warning	Academic Probation
Up to 30	1.99 to 1.51	1.50 to 1.21
31 to 60	1.99 to 1.61	1.60 to 1.21
61 to 90	1.99 to 1.81	1.80 to 1.21
More than 90	1.99 to 1.91	1.90 to 1.21

A student is placed on a fail and discontinue status under any of the following: if less than half the attempted semester credits are passed; if the CGPA is less than 1.21; if two consecutive academic probations are accrued; if any combination of three consecutive academic warnings and/or academic probations are accrued; or fails a course three times even if the CGPA is above 2.00. Such a student is barred from further registration for one semester.

A student is placed on a fail and exclude (FE) status if the student has had two fail and discontinue status, or has been unsuccessful in two programmes. Such a student is barred from further registration for four semesters. A flowchart of the above regulations is shown in Figure A2a and A2b in the Appendix section.

METHODOLOGY

This work used the existing students' academic results for the period August 2002 to December 2007 obtained from the university's student information system. In addition, information concerning courses such as course codes, names, and corresponding credits was obtained from the students' information system. At the end of semester one, the total number of students in the Faculty of Engineering and Technology was 1439, composed of 298 certificate, 443 diploma,

and 698 degree students. Of the total number of students, 348 students were admitted into programmes in August 2007 under the revised regulations (University of Botswana 2007).

The 2007 revised semester-to-semester progression regulations were obtained from the 2007/08 UB Calendar and converted into flowcharts to form the basis for a software program to implement the regulations. The author designed a C' program using Microsoft Visual C++ 6.0 (Microsoft, 1998) to give a software implementation of the revised semester-to-semester progression regulations.

The academic results of students in the Faculty of Engineering and Technology was applied to a computer program that provided the progression recommendations. Although the 2007 regulations were for students admitted in August 2007, the regulations were also applied to the students registered prior to August 2007 to illustrate how the program would handle students who would be in the university for a longer period. The results obtained by the computer program were compared with those obtained using a manual system to verify the correctness of the author's information system based implementation.

SOFTWARE IMPLEMENTATION

The 2007 semester-to-semester progression regulations can be implemented via eight functions to be performed by the software, namely *progress*, *academicWarning*, *academicProbation*, *failDiscontinue*, *failExclude*, *anyFailedCourseThrice*, *detectConsecutive*, and *main*. The flowcharts describing the progression regulations given in Appendix A2 were converted into a pseudo-code description of the computer program given in Appendix A3 to form the basis of the software implementation.

Procedure *progress* determines the academic status of a student at the end of each semester. The procedure checks for any course which is failed more that three times using procedure *anyFailedCourseThrice*. A list of failed courses is updated whenever a failed course is found. Each passed course is checked against the list of failed courses so that it can be removed from the list. If a course is failed three times a status of fail discontinue is placed on the student.

Procedure *progress* then checks if the student has passed at least half the attempted semester credits and places a student who fails the test on a fail and discontinue status via procedure *failDiscontinue*. Procedure *failDiscontinue*, handles the updating of the number of the fail and discontinue status a student has. When this number equals two, a fail and exclude status is applied via procedure *failExclude*. Both the fail and discontinue and fail and exclude status are given a credit limit of zero to indicate that the student cannot register in the subsequent semester.

For a student proceeding on good academic standing, procedure *progress* checks for a CGPA of at least 2.00. For this student a check is made for any failed courses and if there are, a credit limit of 18 otherwise a credit limit of 21 is applied.

For a student proceeding on academic warning or academic probation, procedure *progress* considers the threshold CGPAs indicated in table 1 to determine whether a status of academic warning or academic probation is applied. This number is dependent on the total number of attempted credits. A credit limit of 16 or 14 is applied to a student on academic warning or academic probation respectively. This is done using procedures *academicWarning* and *academicProbation* respectively.

Procedure *progress* also checks for any consecutive academic probations or any combination of three consecutive academic warnings and/or academic probations. To facilitate for this check all previous semester recommendations are logged into a past remark array. The detection of two consecutive academic probations or any combination of three consecutive of academic warning and/or academic probation is implemented using a finite state machine (Allworth, 1981) in procedure *detectConsecutive*. The procedure returns a true value if the required consecutive combinations are found.

For each student, the procedure *main* produces a summary of results printed on a semester basis. The semester results show course results giving the number of credits attempted and the number of credits earned. At the end of semester results the total number of credits attempted, total credits earned and a recommendation for the semester is given. At the end of all semesters a summary result indicating the total attempted credits, earned credits, CGPA, recommendation and credit limit for a student's subsequent semester registration is given.

RESULTS

Samples of results for selected students are shown in Tables 2 to 4. The tables show students' results for various courses grouped on semester basis with the 2007 regulations applied. Against each course, the course credits and credits earned are given. A summary of each semester result is given after the course listing. This summary gives the attempted semester credits (AC), earned credits (EC), cumulative grade point average (CGPA) and the recommendation. Also given is an overall result that includes all attempted credits and a credit limit (CL) applicable to the student.

Table 2 shows results four students with proceed on good academic standing (P), academic warning (AW), academic probation (AP) and a fail and discontinue (FD) status respectively. Given that the total number of attempted credits is less than 31, the applicable CGPA range for AW and AP are 1.99 to 1.51 and 1.50 to 1.21 respectively. The first student proceeds on good academic standing having passed all the courses. The second student proceeds with the AW since the CGPA is 1.60. The student carries a credit limit of 16 for the AW status. The third student proceeds with an AP since the CGPA is 1.30. The student carries a credit limit of 14 for AP status. Note that the above students have passed more than half the attempted semester credits. The fourth student is put on an FD status on account of passing less than half the attempted credits. The earned credits (6) are less than half the attempted semester credits (15) producing an FD result.

Table 2: Proceed, Academic Warning, Academic Probation and Fail and Discontinue Status

20070XXX1	STUDENT 1	20070XXX3	STUDENT 3
2007 Semester 1		2007 Semester 1	
DTB311 B	2 2	EEC111 D-	3 0
DTB312 A	2 2	EEC112 D-	2 0
DTB313 A	2 2	EEC113 C-	2 2
DTB314 B-	3 3	EEC116 E	2 0
DTB315 A	2 2	GEC111 C	2 2
DTB317 B+	2 2	GEC121 C-	2 2
EDT311 B+	2 2	SMC118 C-	2 2
GEC121 A	2 2	----- AC 15 EC 8 CGPA 1.30 Result: AP	
----- AC 17 EC 17 CGPA 4.50 Result: P		----- AC 15 EC 8 CGPA 1.30 Result: AP CL 14	
----- AC 17 EC 17 CGPA 4.50 Result: P CL 21		20070XXX4	STUDENT 4
20070XXX2	STUDENT 2	2007 Semester 1	
2007 Semester 1		EEC212 C	3 3
EEC212 C-	3 3	EEC213 C-	3 3
EEC213 C	3 3	EEC215 D	2 0
EEC215 D	2 0	EEC216 E	3 0
EEC216 D+	3 0	MCC211 D+	2 0
MCC211 C-	2 2	SMC218 D-	2 0
SMC218 E	2 0	----- AC 15 EC 6 CGPA 1.30 Result: FD	
----- AC 15 EC 8 CGPA 1.60 Result: AW		----- AC 15 EC 6 CGPA 1.30 Result: FD CL 0	
----- AC 15 EC 8 CGPA 1.60 Result: AW CL 16			

To illustrate how the program would process results of students who have been in the university for more than a semester, results of students admitted prior to August 2007 are used. Such students are currently covered by the 2002 regulations. For these students, Tables 3 and 4 show results obtained when the revised regulations are applied using the program.

Table 3: Proceed Status

912XXX1	STUDENT 1	GEC122 A	2	2	
2006 Semester 1		ITB200	4	0	
CCB211 C+	2	2	MAT292 D	3	0
CCB212 C-	2	2	MMB221 A	2	2
DTB211 B	2	2	MMB222 C-	2	2
EEB211 B	2	2	----- AC 21 EC 14 CGPA 2.87 Result: P		
GEC111 B+	2	2	2007 Semester 1		
GEC121 A	2	2	CBB311 C-	3	3
MAT291 C-	3	3	CCB312 A	2	2
MMB211 B-	2	2	CCB313 B-	3	3
----- AC 17 EC 17 CGPA 3.41 Result: P		CCB315 D+	2	0	
2006 Semester 2		GEC257 B+	2	2	
CCB221 C-	2	2	MAT391 D+	3	0
DTB221 C+	2	2	MGT100 B	3	3
EEB221 C	2	2	----- AC 18 EC 13 CGPA 2.93 Result: P		
GEC112 B+	2	2	----- AC 56 EC 44 CGPA 2.93 Result: P CL 18		

Table 4: Fail and Discontinue/Fail and Exclude Status

950XXX1 STUDENT 1	2005 Semester 1
2003 Semester 1	GEC121 B+ 2 2
CCB211 B 2 2	MMB311 B- 3 3
CCB212 D+ 2 0	MMB411 C+ 2 2
DTB211 B- 2 2	MMB412 B 2 2
EEB211 D+ 2 0	MMB413 C- 3 3
GEC257 B- 2 2	MMB415 B 2 2
MAT291 C- 3 3	----- AC 14 EC 14 CGPA 2.85 Result: P
MMB211 C- 2 2	2005 Semester 2
----- AC 15 EC 11 CGPA 2.53 Result: P	CCB221 E 2 0
2003 Semester 2	GEC122 E 2 0
CCB221 D+ 2 0	MMB222 E 2 0
DTB221 B 2 2	MMB324 E 3 0
EEB221 B- 2 2	MMB522 E 3 0
GEC247 A 3 3	MMB524 E 4 0
ITB200 C+ 4 4	----- AC 16 EC 0 CGPA 2.39 Result: FD FE 3F-CCB221
MAT292 B 3 3	2006 Semester 1
MMB221 B+ 2 2	MMB313 C+ 3 3
MMB222 D 2 0	MMB511 B 3 3
----- AC 20 EC 16 CGPA 3.03 Result: P	MMB512 B+ 3 3

2004 Semester 1	MMB513 B 4 4
CCB212 A 2 2	MMB524 C+ 4 4
EEB211 C+ 2 2	----- AC 17 EC 17 CGPA 2.58 Result: P
MAT391 C 3 3	2006 Semester 2
MMB312 B- 2 2	GEC122 C+ 2 2
MMB314 B- 2 2	MMB323 C- 3 3
MMB414 B- 3 3	MMB324 C 3 3
MMB418 C+ 2 2	MMB521 3 0
----- AC 16 EC 16 CGPA 3.14 Result: P	MMB522 B+ 3 3
2004 Semester 2	MMB523 A 4 4
CCB221 E 2 0	----- AC 18 EC 15 CGPA 2.63 Result: P
EEB326 C- 3 3	2007 Semester 1
MMB222 D- 2 0	GEC111 E 2 0
MMB322 C- 2 2	ITB420 10 0
MMB323 D+ 3 0	----- AC 12 EC 0 CGPA 2.41 Result: FD FE
MMB324 D 3 0	----- AC 145 EC 96 CGPA 2.41 Result: FE CL 0
MMB325 B 2 2	
----- AC 17 EC 7 CGPA 2.74 Result: FD	

Table 3 shows the result of a student who has been in the university for three semesters. The student has an overall CGPA of 2.93, and proceeds on good academic standing with a P status. However, a credit limit of 18 is imposed on account of having failed course CCB315.

Table 4 shows results for a student who joined in the second year of a five-year degree programme. In the first three semesters the student progresses with a P status. In 2004 semester 2, the student is put on an FD status on account of having passed less than half the attempted

semester credits. In semester 6, the student failed CCB221 for the third time. The student is put on FD on account of having passed less than half the attempted semester credits and failing a course (CCB221) three times. Note that now the student has acquired two FD status which results into an FE status. At the end of semester 9, the student had accumulated three FD status leading to an FE. Overall the student attempted 145 credits, earned 96 credits, a CGPA of 2.41 and a credit limit of 0.

DISCUSSION

The results show that the academic recommendations for the students arrived at using the program are in agreement with the expected ones (Tables 2 and 3). The results of a student who joined in the second year of a five-year degree programme under the 2002 regulations are shown in Table 4. These recommendations have been arrived at using the 2007 regulations. Since this student is covered by 2002 regulations, they were applied to allow us to compare the results.

At the end of 2004 semester 2, the student had at least a failed core course in 2003 semester 1 (CCB212 and EEB211), 2003 semester 2 (CCB221 and MMB221), and in 2004 semester 2 (CCB221). Under the 2002 regulations the student should have had acquired three probationations at this point, leading to an FD. However, the student progressed to the next semester as the system in use at the time could not detect the above condition for an FD to be applied. At the end of 2005 semester 2, the student would have acquired another FD on account of having failed all the courses. This condition was also not detected and the student progressed to the next semester. A similar situation occurs at the end of semester 9. Note that at the end of the semester, the student's CGPA was above 2.0; therefore if emphasis was placed on checking for CGPA below 1.5 the FD conditions could easily be missed. Note also that the application of the program would have detected the FD conditions as illustrated in Table 4.

The implementation of the semester-to-semester progression using a computer program produced expected results status, CGPA, attempted credits, and earned credits. When used on students covered by 2002 regulations the consolidated results obtained allow easier manual checks to be performed.

For the status of FD and FE, in our implementation a student carried a credit limit of zero. In practice, the student's information system can be programmed to apply an appropriate code which would result in blocking a student from registration for a period of one semester or four semesters for an FD or FE respectively.

The implementation also identified two issues not handled by the regulations; how to handle a student who returns to the programme after an FE status or a student who returns from an FD status for having failed the same course three times. The student returning from an FE has at least two FD status on their academic record; similarly a student returning from an FD for having failed the same course three times has the three courses on their academic record. Should these be excluded in determining the student's status in the following semesters thereafter? The situation is handled in other universities by using a terminal FE (University of Brighton, 2007) or for academic dismissal (Kutztown University, 1994); a student cannot be readmitted in the same programme. The Oklahoma State University (Oklahoma State University, 2007) allows a student

to be readmitted only once but also caters for a student to take courses in another institution to remediate the failure cause before readmission. However, a number of universities have regulations that allow students to be readmitted more than once after an FD status such as at New Mexico State University (New Mexico State University, 2007). The university could consider adopting the terminal FE mode to handle the case of a student who has two FD status if the student was to be readmitted. The same mode can be used to handle the case of a student who fails the same course three times. Nevertheless, if it is desirable to readmit students after an FE status such students should be handled administratively since regulations do not cover them. This work takes the latter approach.

CONCLUSION

The transition from a yearlong to a semester academic system required a revision of regulations including the semester-to-semester progression. The 2002 regulations were implemented without being subjected to scrutiny across faculties. It soon became apparent that the regulations had omissions and inconsistencies. This made them difficult to implement fully on the students' information system or via a manual system and necessitated a revision which resulted in the 2007 regulations. The revised regulations now require that the information system be updated to enable a complete and uniform application for all students.

The paper has shown that the semester-to-semester progression regulations at the University of Botswana can be implemented via a computer program. The implementation by the author covered main aspects of the regulations whose application was demonstrated on a sample of students' results at the University of Botswana in the Faculty of Engineering and Technology. The results generated were found to be in agreement with the expected academic recommendations when compared with those obtained using a manual system. The program code can be adapted to the information system at the university.

The advantage of the suggested system is that it saves time and effort in providing students' academic recommendations at the end of a semester. In addition, the system provides a uniform application of the semester-to-semester progression regulation to all students. When the system is applied to students following the 2002 regulations, the system provides consolidated information which makes manual checks easier.

Uniform application of progression regulations require an implementation that is based on an information system. However, the implementation can be problematic, since the vendor of the information system may not be prepared to incorporate the progression regulations or their revisions due to a lack of comprehension of the regulations. In cases where vendors are reluctant to implement the regulations on an information system an alternative solution must be implemented in good time to avoid the use of a manual system for an extended period. Most universities have an Information Technology, Computer Science or other Engineering departments which could be tasked to provide an in house alternative implementation as shown in this paper. In addition, revision of regulations should include an assessment of the suitability of the proposed regulations for implementation on an information system.

REFERENCES

University of Botswana, (2006). *University of Botswana 2006-2007 Calendar*, Public Affairs Department Publication Unit, pp 11-15, Gaborone.

University of Botswana, (2007). *University of Botswana 2007-2008 Calendar*, Public Affairs Department Publication Unit, pp 13-14, Gaborone.

Microsoft, (1998). *Microsoft Visual C++ 6.0, Microsoft Visual C++ User's Guide*, [http://msdn.microsoft.com/en-us/library/aa232254\(VS.60\).aspx](http://msdn.microsoft.com/en-us/library/aa232254(VS.60).aspx).

Allworth, S.T., (1981), *Introduction to Real-time Software Design*, Macmillan Computer Science Series, pp 100-109, London.

University of Brighton, (2007), *University of Brighton: General Examination and Assessment Regulations for Taught Courses (2007/08)*, <http://staffcentral.brighton.ac.uk/xpedio/groups/Public/documents/staffcentral/doc001110.pdf>.

Kutztown University, (1994), *Kutztown University Policy 1994-001-A*, <http://www.kutztown.edu/admin/AdminServ/academic/acad001.html>.

Oklahoma State University, (2007), *2007-08 University Academic Regulations*, pp 58, Transcript Press, Stillwater, Oklahoma, <http://www.okstate.edu/registrar/Catalogs/2007-2008/Catalog2007-2008.pdf>.

New Mexico State University, (2001), *New Mexico State University Undergraduate Catalog*, Vol. 95, No. 3, June 2001, pp 22, New Mexico.

APPENDIX A1

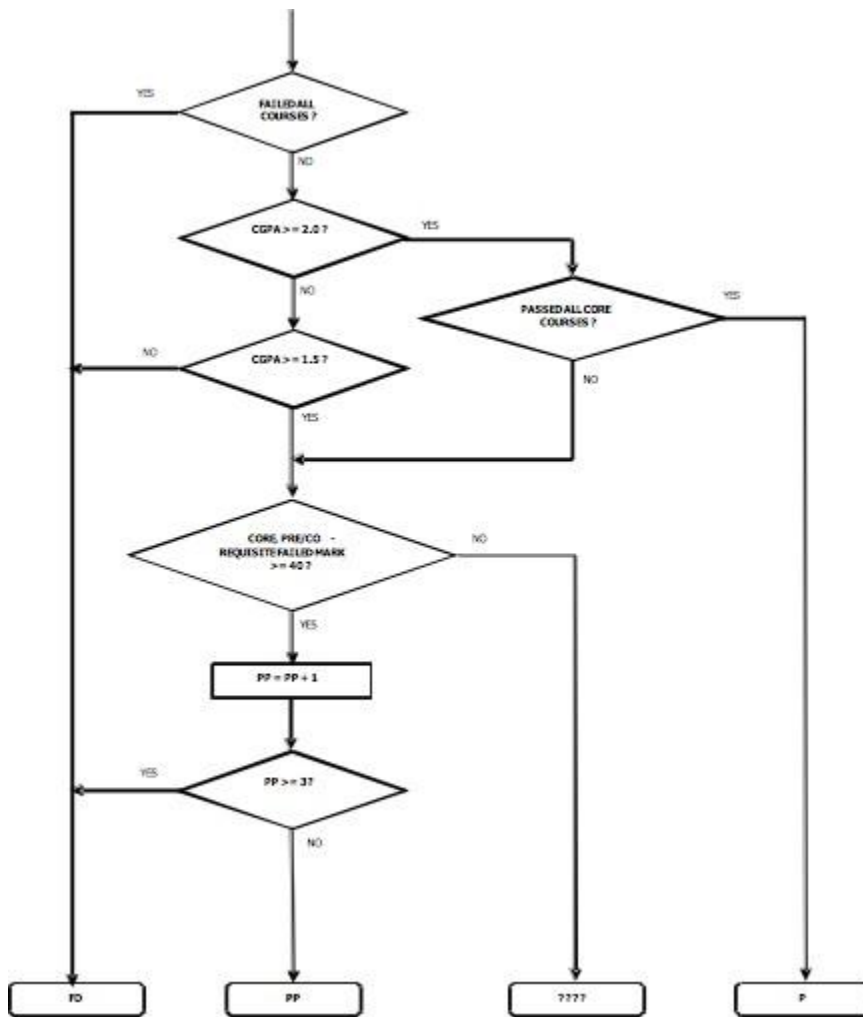


Figure A1: Flowchart Progression – 2002 Regulation

APPENDIX A2

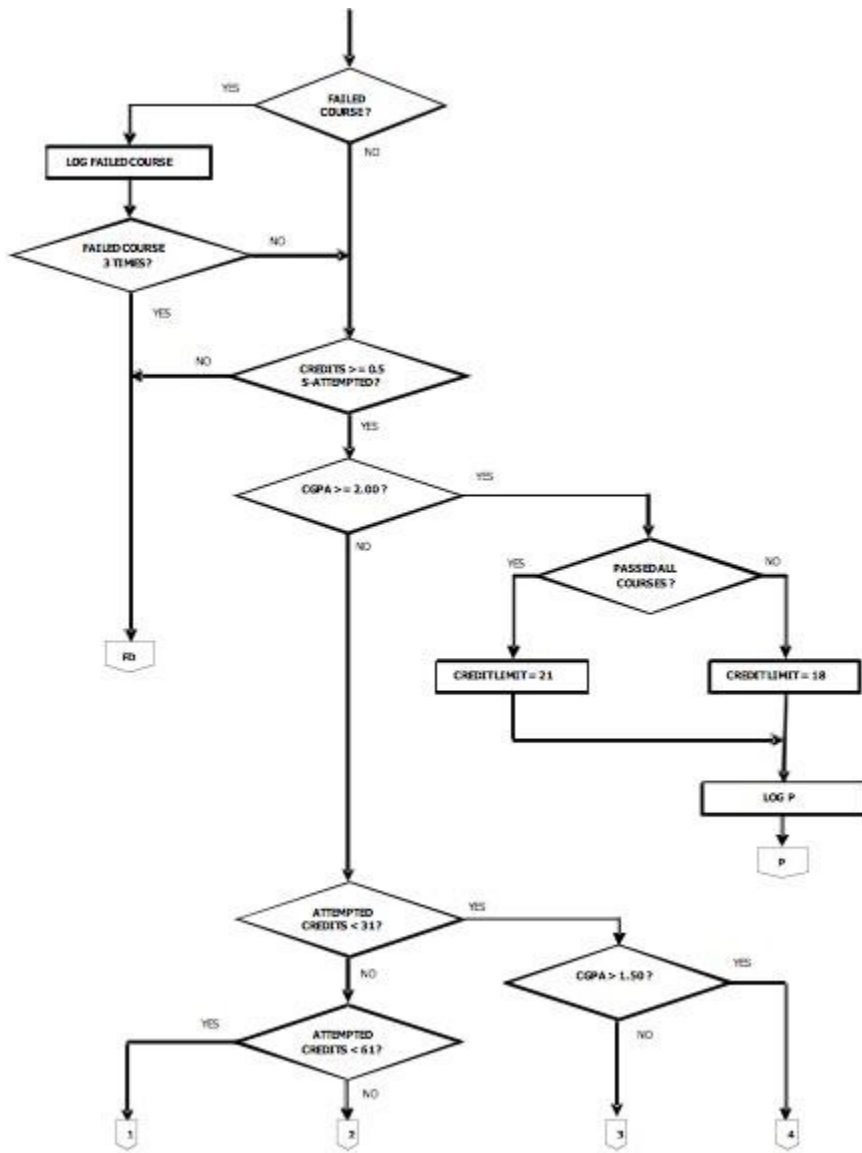


Figure A2a: Flowchart Progression – 2007 Revised Regulation

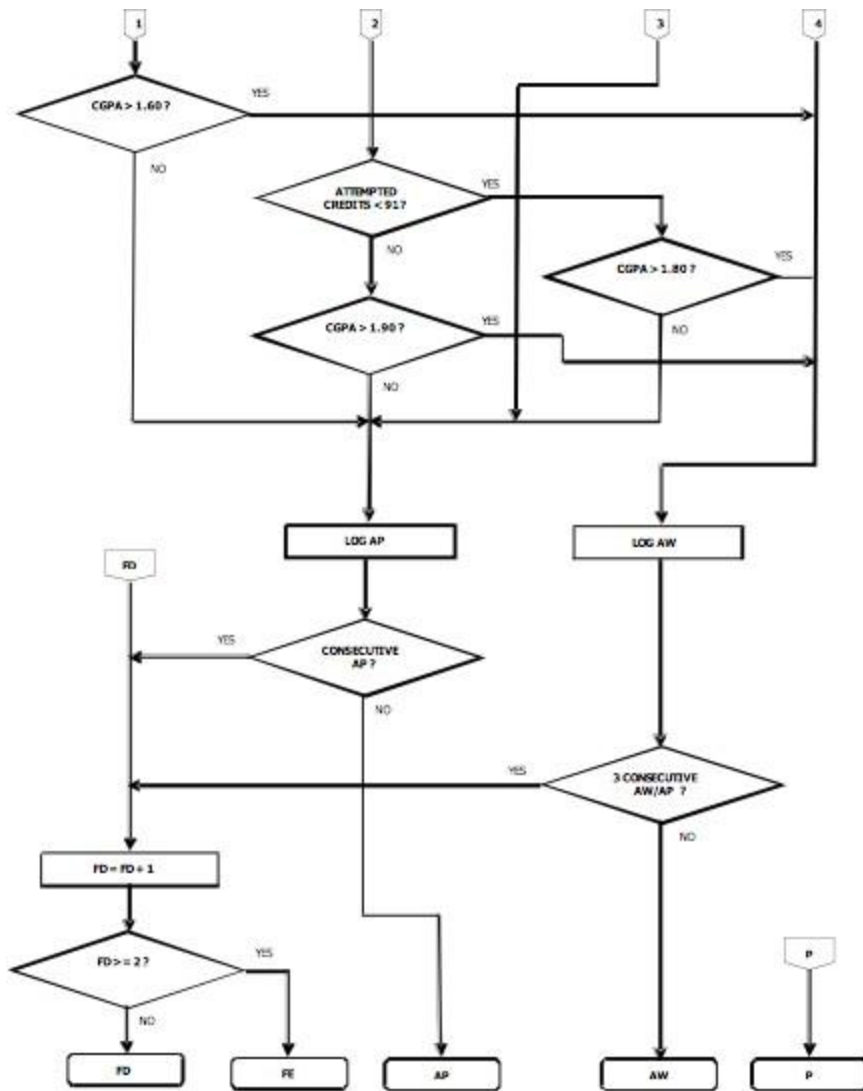


Figure A2b: Flowchart Progression – 2007 Revised Regulation

APPENDIX A3

Program Description (Pseudo-Code)

The implementation pseudo-code is based on the flowcharts given in Appendix A2.

procedure progress()

```
if (anyFailedCourseThrice()) {
    failDiscontinue();
    return;
}
if (semesterEarnedCredits*2 > semesterAttempted Credits) {
    failDiscontinue();
    return;
}
if (attemptedCredits > 0) {
    tempcGPA=cGPA/attemptedCredits;
} else {
    failDiscontinue();
    return;
}
if (tempcGPA >= 2.00) {
    if (semesterAttemptedCredits == semesterEarnedCredits) { // any failed courses?
        creditLimit=21;
    } else {
        creditLimit=18;
    }
    proceed(P);
    return;
}
academicStatus=AS_AP;
if (attemptedCredits < 31 && tempcGPA > 1.50) { // table 1 academic warning/probation
    academicStatus=AS_AW;
} elseif ( attemptedCredits <61 && tempcGPA > 1.60) {
    academicStatus=AS_AW;
} elseif (attemptedCredits <91 && tempcGPA > 1.80) {
    academicStatus=AS_AW;
} elseif (tempcGPA > 1.90) {
    academicStatus=AS_AW;
}
if (academicStatus == AS_AW) {
    adademicWarning();
} else {
    academicProbation();
}
```


procedure academicWarning()

```
proceed(AW);
creditLimit=16;
pastRemark[count++]=AS_AW;
if (detectConsecutive(pastRemark)) {
    failDiscontinue();
    return;
}
print(" AW ");
```

procedure academicProbation

```
proceed(AP);
creditLimit=14;
pastRemark[count++]=AS_AP;
if (detectConsecutive(pastRemark)) {
    failDiscontinue();
    return;
}
print(" AP ");
```

prodecure failDiscontinue()

```
print(" FD ");
++fdCount;
if (fdCount >= 2) {
    failExclude();
    return;
}
```

Procedure failExclude()

```
pastRemark[cCount++]=AS_FE;
creditLimit=0;
print(" FE ");
```

procedure anyFailedCourseThrice()

```
search failed course list for any course appearing three times;
if (found) {
    return location;
} else {
    return null;
}
```

procedure detectConsecutive()

```
state=0;
for (each remark) {
    switch(state) {
        case 0:
```

```

    if (remark==AS_AP) state=1;
    else if (remark==AS_AW) state=2;
    else state=0;
    break;
case 1:
    if (remark==AS_AP) return 1;
    else if (remark==AS_AW) state=3;
    else state=0;
    break;
case 2:
    if (remark==AS_AP//remark==AS_AW) return 1;
    else if (remark==AS_AW) state=5;
    else state=0;
    break;
case3:
case4:
case5:
    if (remark==AS_AP//remark==AS_AW) return 1;
    else state=0;
    break,
}
return 0;
}

```

procedure main()

```

for each student's result {
    earnedCredits=0;
    attemptedCredits=0;
    cGPA=0;
    initialise past remark array;
    for( each semester completed) {
        semesterAttemptedCredits=0;
        semesterEarnedCredits=0;
        for each semester course {
            identify cCode, cCredits, cGrade, cGradePoint;
            cGPA=cGPA+cCredits*cGradePoint;
            attemptedCredits=attemptedCredits+cCredits;
            semesterAttemptedCredits=semesterAttemptedCredits+cCredits;
            if (cGrade==passingGrade) {
                semesterEarnedCredits=semesterEarnedCredits+cCredits;
                earnedCredits=earnedCredits+cCredits;
                deleteFailedCourse(cCode);
            } else {
                addFailedCourse(cCode);
            }
        }
    }
}

```

```
    print(attemptedCredits, earnedCredits, cGPA/attemptedCredits);  
    progress();  
    }  
}  
cGPA=cGPA/attemptedCredits;  
print(attemptedCredits, earnedCredits, cGPA, creditLimit);  
}
```