**Computer Science Assignment: Investigation stage**

**Problem Description:** Ms Anita Kakar wants a software which can calculate the predicted atar score for year 11 and 12 students in atar, calculate the number of Cs they have earned so far, determine and display whether the student has passed OLNA or not, calculate the GPA for general students and also the amount of Cs which they have currently earned. It should also check whether they have earned the Cs from courses like Applied Islam and VET courses.

If a student does not accumulate 8 Cs over the course of year 11 and 6 Cs over the course of year 12 or 14 Cs altogether, they cannot graduate. If a student does not pass OLNA, they also do not pass. The spreadsheet should also display the subjects for all students and their marks.

For year 7-10 students, it should calculate the GPA of the students by taking the average of the main 4 subjects. The 4 main subjects for all year 7-10 students are Mathematics, English, Science and HASS (Humanities and Social Sciences).

**Requirements:**

1.) Calculate the number of Cs for year 11 and 12 students

2.) Determine whether the student has passed OLNA or not

3.) Calculate predicted atar

4.) Determine whether students who are doing year long courses such as VET or applied Islam.

5.) Calculate the GPA for general students

6.) Calculate the GPA for year 7-10 students

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Feature** | **Task** | **Estimated Time** | **Dependencies** | **Functional or Non-Functional** |
| **Calculating the Number of Cs** | **1.)** Determine what subjects a student is currently taking. | 1-2 hours per cohort | None | **Functional** |
|  | 2.) Determine the amount of Cs he is accumulating for each subject | 2-3 hours per cohort | Task 1 | **Functional** |
|  | 3.) Plot the amount of Cs on the spreadsheet | 1-2 hours per cohort | Task 2 | **Functional** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Feature** | **Task** | **Estimated Time** | **Dependencies** | **Functional or Non-Functional** |
| **Determine whether the student has passed OLNA or not** | 1.) Check on TASS whether the student has met OLNA requirements | 1-2 hours per cohort | None | **Functional** |
|  | 2.) Import the response into the spread sheet | 1-2 hours per cohort | Task 1 | **Functional** |
|  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Feature** | **Task** | **Estimated Time** | **Dependencies** | **Functional or Non-Functional** |
| **Calculating the Predicted Atar** | 1.) Determine what subjects a student is taking | 1-2 hours per cohort | None | **Functional** |
|  | 2.) Determine the marks they are accumulating for all the subjects. | 2-3 hours per cohort | Task 1 | **Functional** |
|  | 3.) Determine the average of the subjects after scaling | 2-3 hours per cohort | Task 2 | **Functional** |
|  | 4.) Add the average of the top 4 subjects after scaling, to determine the TEA. | 1-2 hours per cohort | Task 3 | **Functional** |
|  | 5.) Plug the TEA into the TISC calculator and then insert that value (predicted atar) into the spreadsheet. |  | Task 4 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Feature** | **Task** | **Estimated Time** | **Dependencies** | **Functional or Non-Functional** |
| **Determining whether students who are doing year long courses have passed or not** | 1.) Check on TASS whether they have passed the subject or not, accumulated 2 Cs. | 1-2 hours per cohort | None | **Functional** |
|  | 2.) Import the response into the spreadsheet. | 1-2 hours per cohort | Task 1 | **Functional** |
|  |  |  |  |  |
|  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Feature** | **Task** | **Estimated Time** | **Dependencies** | **Functional or Non-Functional** |
| **Calculating the GPA for general students** | 1.) Determine the main top 4 subjects for the student | 2-3 hours per cohort | None | **Functional** |
|  | 2.) Calculate the average by adding the final average for each subject then divide it by 4 | 1-2 hours per cohort | Task 1 | **Functional** |
|  | 3.) Import the response into the spreadsheet as the GPA. | 1-2 hours per cohort | Task 2 |  |
|  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Feature** | **Task** | **Estimated Time** | **Dependencies** | **Functional or Non-Functional** |
| **Calculating the GPA for year 7-10 students.** | 1.) Determine the average for the mean 4 subjects, by adding the final averages together and dividing it by 4. | 2-3 hours per cohort | None | **Functional** |
|  | 2.) Import the value into the spreadsheet as the GPA | 1-2 hours per cohort | Task 1 | **Functional** |
|  |  |  |  |  |
|  |  |  |  |  |

**Pseudocode**

MAIN MODULE

// Function to calculate GPA for Year 7-10 students

Function calculate\_GPA(subjects):

// subjects is a list of grades for 4 main subjects: Mathematics, English, Science, HASS

sum\_of\_grades = 0

for each subject in subjects:

sum\_of\_grades = sum\_of\_grades + subject.grade

GPA = sum\_of\_grades / 4

RETURN GPA

// Function to calculate the number of Cs earned by the student

Function calculate\_number\_of\_Cs(grades):

number\_of\_Cs = 0

for each grade in grades:

if grade == 'C':

number\_of\_Cs = number\_of\_Cs + 1

RETURN number\_of\_Cs

// Function to check if student passed OLNA

Function check\_OLNA\_status(OLNA\_result):

IF OLNA\_result == "Pass":

RETURN True

ELSE:

RETURN False

// Function to check if the student can graduate (for Year 11 and 12 students)

Function check\_graduation\_status(year, grades, OLNA\_result, VET\_courses):

number\_of\_Cs = calculate\_number\_of\_Cs(grades)

passed\_OLNA = check\_OLNA\_status(OLNA\_result)

// Check conditions for graduation

IF year == 11 or year == 12:

IF (number\_of\_Cs >= 8 in Year 11) OR (number\_of\_Cs >= 6 in Year 12) OR (number\_of\_Cs >= 14 in total):

IF passed\_OLNA == True:

RETURN "Graduation Status: Passed"

ELSE:

RETURN "Graduation Status: Failed (OLNA)"

ELSE:

RETURN "Graduation Status: Failed (Not enough Cs)"

ELSE:

RETURN "Graduation Status: Not applicable (not Year 11 or 12)"

// Function to check if a student has earned Cs from Applied Islam or VET courses

Function check\_Cs\_in\_special\_courses(grades, special\_courses):

special\_courses\_with\_Cs = []

for each course in special\_courses:

IF course.name in grades AND grades[course.name] == 'C':

special\_courses\_with\_Cs.append(course.name)

RETURN special\_courses\_with\_Cs

// Main function for calculating and displaying student data

Function main():

// Input student data

student = get\_student\_data() // Fetches student data

year = student.year

subjects = student.subjects // List of all subjects and their grades

OLNA\_result = student.OLNA\_result // OLNA result: 'Pass' or 'Fail'

special\_courses = student.special\_courses // List of Applied Islam and VET courses

// Display all subjects and marks

PRINT "Subjects and Marks for ", student.name

for each subject in subjects:

PRINT subject.name, ": ", subject.grade

// Calculate GPA for Year 7-10

IF year >= 7 AND year <= 10:

GPA = calculate\_GPA([Mathematics, English, Science, HASS])

PRINT "GPA: ", GPA

// Calculate number of Cs and check graduation status for Year 11-12

IF year >= 11 AND year <= 12:

graduation\_status = check\_graduation\_status(year, subjects, OLNA\_result, special\_courses)

PRINT graduation\_status

// Check if the student earned Cs in Applied Islam or VET courses

special\_courses\_with\_Cs = check\_Cs\_in\_special\_courses(subjects, special\_courses)

IF special\_courses\_with\_Cs:

PRINT "Cs earned in special courses: ", special\_courses\_with\_Cs

ELSE:

PRINT "No Cs earned in special courses"

ENDIF

END Function

// Execute the main function

main()

END MODULE