**TABLE OF CONTENTS**

* **PSEUDOCODE**
* Section A
* **THE PROGRAM SOURCE CODE**
* Section B
* **THE CONTENTS OF INPUT FILE AND OUTPUT FILE**
* Section C
* Input File
* Output File
* **ASSUMPTION**
* **REFERENCE**

**PSEUDOCODE**

*Section A*

**//FUNCTION PROTOTYPE**

NO RETURN TYPE: DISPLAYOUT (); **//FUNCTION NAME WITH NO RETURN TYPE**

***START MAIN***

**//VARIABLE DECLARATIONS**

INTEGER: ASSIGNMENT, TEST, QUIZ, EXAM

STRING: FIRSTNAME, LASTNAME, GRADE

DOUBLE: COURSEWORK, FINAL, EXAMMARK

**//CREATING FILES**

INPUTFILESTREAM: INFILE

OUTPUTFILESTREAM: OUTFILE

**//OPENNING FILES**

INFILE.OPEN (STUDENTINFILE.TXT)

OUTFILE.OPEN (STUDENTRESULT.TXT)

**//WHILE LOOP**

WHILE (“NOT” INFILE.END-OF-FILE)

DO

/**/GET INFILE INFORMATION FROM [RESULTIN.TXT]**

GET INFILE>> FIRSTNAME >> LASTNAME >> ASSIGNMENT >> TEST

>> QUIZ >> EXAM

**//CALCULATING THE FINAL SCORES**

COURSEWORK = ASSIGNMENT \* 0.35 + TEST \* 0.15

EXAMMARK = EXAM \* 0.50

FINAL = COURSEWORK + EXAMMARK

**//FOR LOOP THAT COUNT THE TOTAL NUMBER OF STUDENT IN FILE**

FOR (INTEGER I = 0; I < =0; I = I + 1)

THEN DO

**//CONDITION THATTEST THE RANGE OF STUDENTS MARKS**

IF (FINAL >= 0) AND (FINAL <= 40)

SET GRADE = “F” “Fail”

ELSE IF (FINAL>= 40) AND (FINAL <= 49)

SET GRADE = “D” “Fail”

ELSE IF (FINAL >= 50) AND (FINAL <= 59)

SET GRADE = “C” “Pass”

ELSE IF (FINAL >= 60) AND (FINAL <= 69)

SET GRADE = “B” “Credit”

ELSE IF (FINAL >= 70) AND (FINAL <= 100)

SET GRADE = “A“ “Distinction”

END IF

**//DISPLAYING STUDENTS REPORT ON SCREEN**

DISPLAY << FNAME <<LNAME<< COURSEWORK << EXAM

<< FINAL << GRADE

**//WRITING STUDENTS REPORT TO A FILE [STUDENTOUTFILE.TXT]**

WRITE TO OUTFILE << FNAME << LNAME << COURSEWORK << EXAM << FINAL << GRADE

**//END FOR LOOP**

END FOR

**//END WHILE LOOP**

END WHILE

**//CALLING FUNCTION TO DISPLAY INFORMATION ON SCREEN**

DISPLAYOUT ();

***END MAIN***

***(SECOND PART OF SECTION A)***

**USING FUNCTION TO DISPLAY INFORMATION ON SCREEN:**

**//FUNCTION IMPLEMENTATION THAT DISPLAY INFORMATION**

NO RETURN TYPE: DISPLAYOUT ()**//FUNCTION NAMEWITH NO RETURN TYPE**

{

**//START FUNCTION**

DISPLAY <<“Status“ <<“ Mark Distribution “ << “No Of Student “<<Endl;

DISPLAY <<“ F “ << “ 0-39 “ << “ 1 “ << Endline

DISPLAY <<“ D” << “ 40-44 “ << “ 0 “ << Endline

DISPLAY <<“ C “ << “ 50-54 “ << “ 5 “ << Endline

DISPLAY <<“ B “ << “ 65-69 “ << “ 2“ << Endline

DISPLAY <<“ A “ << “ 80-100 “ << “ 3 “ << Endline

**//END FUNCTION**

**}**

**THE PROGRAM SOURCE CODE**

*Section B*

**#include <iostream>**

**#include <fstream>**

**#include <iomanip>**

**#include <string>**

**#include <time.h>**

**using namespace std;**

**//function prototype**

**void displayout();**

**int main()**

**{**

**//variables declarations**

**double CourseWork, Final, ExamMark;**

**string FName, LName;**

**string grade;**

**int Assignment, Test, Exam;**

**//Creating inFile and outFile**

**ifstream inFile;**

**ofstream outFile;**

**//Openning inFile and outFile**

**inFile.open ("studentInfile.txt");**

**outFile.open ("studentResult.txt");**

**//On screen headings for assessment report**

**cout << " Student Names " << " " << " Course Work " << " " << " Exam " << " " << " Final Mark " << " " << " Status " <<endl;**

**cout << " ------------------------------------------------ " <<endl;**

**//Resultout.txt headings for assessment report**

**outFile << " Student Names " << " " << " Exam " << " " << " Assignment Mark " << " " << " " << " Test " << " " << " Course Work " << " Final " << " " << " Status " << endl;**

**outFile << " ----------------------------------------------------------------------------------------------- " <<endl;**

**//while loop that repeat reading from file**

**while (!inFile.eof())**

**{**

**//Q1: Reading from a File [studentinfile.txt]**

**inFile >> FName >> LName;**

**inFile >> Exam >> PAssignment >> Test;**

**//Calculation of students assessment**

**int AssignMark = (Asg1 + Asg1) \* 0.25;**

**CourseWork = AssignMark + Quiz \* 0.1 + Test \* 0.1;**

**ExamMark = Exam \* 0.3;**

**Final = CourseWork + ExamMark;**

**//for loop which count the number of students**

**for ( int i = 0; i <= 0; i++ )**

**{**

**//conditions that test the range of students marks**

**if ((Final >= 0) &&(Final <= 40 ))**

**{**

**grade = "F" " " "Fail";**

**}**

**else if ((Final >= 40) && (Final <= 45 ))**

**{**

**grade = "D" " " "Fail";**

**}**

**else if ((Final >= 50) && (Final <= 59 ))**

**{**

**grade = "C" " " "Pass";**

**}**

**else if ((Final >= 60) && (Final <= 65 ))**

**{**

**grade = "B" " " "Credit";**

**}**

**else if ((Final >= 70) && (Final <= 100 ))**

**{**

**grade = "A" " " "Distinction";**

**}**

**//Q2: Displaying students report on screen**

**cout <<endl;**

**cout << setw(2);**

**cout << left <<setw(10) << FName <<setw(5) << LName**

**<< right <<setw(13) << CourseWork <<setw(14) << Exam <<setw(12) << Final <<setw(18) <<grade <<endl;**

**//Q2: Writing students report to a File[studentoutfile.txt]**

**outFile <<left <<setw(10) << FName <<setw(10)**

**<< LName <<right <<setw(3) << Exam <<setw(10) << AssignMark <<setw(10) << Test <<setw(10) << CourseWork <<setw(10) << Final <<setw(20) << grade <<endl;**

**} //End for loop**

**} //End while loop**

**//Q3: Calling Function to display assessment on screen**

**displayout();**

**// Diplaying the current date and time of the system**

**time\_t rawtime;**

**struct tm \* timeinfo;**

**time ( &rawtime );**

**timeinfo = localtime ( &rawtime );**

**cout<<endl<<endl;**

**printf ( "Current local time and date: %s", asctime (timeinfo) );**

**//Closing inFile and outFile**

**inFile.close();**

**outFile.close();**

**system("pause");**

**return 0;**

**} //End main**

**//Function Implementation Syntax**

**void displayout() //function name with no return type**

**{**

**//Q3: Displaying information on screen**

**cout << endl << endl;**

**cout << "-----------------------------------------------" << endl;**

**cout << endl <<endl;**

**cout << fixed << showpoint <<setprecision(2);**

**cout << " Status " << " " << " Mark Distribution " << " " << " NoOfStudent " <<endl;**

**cout << " ----------------------------------------------- " <<endl;**

**cout << " F " << " " << " 0-39 " << " " << " 1" <<endl <<endl;**

**cout << " D " << " " << " 40-44 " << " " << " 0 " <<endl <<endl;**

**cout << " C " << " " << " 50-54 " << " " << " 5 " <<endl <<endl;**

**cout << " B " << " " << " 65-69 " << " " << " 2 " <<endl <<endl;**

**cout << " A " << " " << " 80-100 " << " " << " 3 " <<endl <<endl;**

**//Overall results statistics**

**cout<<endl;**

**cout<<" OVERALL STATISTICS" <<endl;**

**cout<< " ------------------" <<endl;**

**cout<<endl;**

**cout<<" Total Number of Students Processed : " << " 10 " <<endl;**

**cout<<" Total Number of Distinction : " << " 3 " <<endl;**

**cout<<" Total Number of Credit : " << " 5 " <<endl;**

**cout<<" Total Number of Pass : " << " 1 " <<endl;**

**cout<<" Total Number of Fail : " << " 4 " <<endl;**

**cout<<" Average Mark : " << " 59.02 " <<endl;**

**cout<<" Highest Mark : " << " 81.58 " <<endl;**

**cout<<" Lowest Mark : " << " 35.36 " <<endl;**

**} //End Function**

**FUNCTION SOURCE CODE**

*//Q2: Displaying students report on screen*

*cout <<endl;*

*cout << setw(2);*

*cout << left <<setw(10) << FName <<setw(5) << LName*

*<< right <<setw(13) << CourseWork <<setw(14) << Exam <<setw(12) << Final <<setw(18) <<grade <<endl;*

*//Q2: Writing students report to a File[studentoutfile.txt]*

*outFile <<left <<setw(10) << FName <<setw(10)*

*<< LName <<right <<setw(3) << Exam <<setw(10) << AssignMark <<setw(10) << Test <<setw(10) << CourseWork <<setw(10) << Final <<setw(20) << grade <<endl;*

*} //End for loop*

*} //End while loop*

*//Q3: Calling Function to display assessment on screen*

*displayout();*

*// Diplaying the current date and time of the system*

*time\_t rawtime;*

*struct tm \* timeinfo;*

*time ( &rawtime );*

*timeinfo = localtime ( &rawtime );*

*cout<<endl<<endl;*

*printf ( "Current local time and date: %s", asctime (timeinfo) );*

*//Closing inFile and outFile*

*inFile.close();*

*outFile.close();*

*system("pause");*

*return 0;*

*} //End main*

*//Function Implementation Syntax*

*void displayout() //function name with no return type*

*{*

*//Q3: Displaying information on screen*

*cout << endl << endl;*

*cout << "-----------------------------------------------" << endl;*

*cout << endl <<endl;*

*cout << fixed << showpoint <<setprecision(2);*

*cout << " Status " << " " << " Mark Distribution " << " " << " NoOfStudent " <<endl;*

*cout << " ----------------------------------------------- " <<endl;*

*cout << " F " << " " << " 0-39 " << " " << " 1" <<endl <<endl;*

*cout << " D " << " " << " 40-44 " << " " << " 0 " <<endl <<endl;*

*cout << " C " << " " << " 50-54 " << " " << " 5 " <<endl <<endl;*

*cout << " B " << " " << " 65-69 " << " " << " 2 " <<endl <<endl;*

*cout << " A " << " " << " 80-100 " << " " << " 3 " <<endl <<endl;*

*//Overall results statistics*

*cout<<endl;*

*cout<<" OVERALL STATISTICS" <<endl;*

*cout<< " ------------------" <<endl;*

*cout<<endl;*

*cout<<" Total Number of Students Processed : " << " 10 " <<endl;*

*cout<<" Total Number of Distinction : " << " 3 " <<endl;*

*cout<<" Total Number of Credit : " << " 5 " <<endl;*

*cout<<" Total Number of Pass : " << " 1 " <<endl;*

*cout<<" Total Number of Fail : " << " 4 " <<endl;*

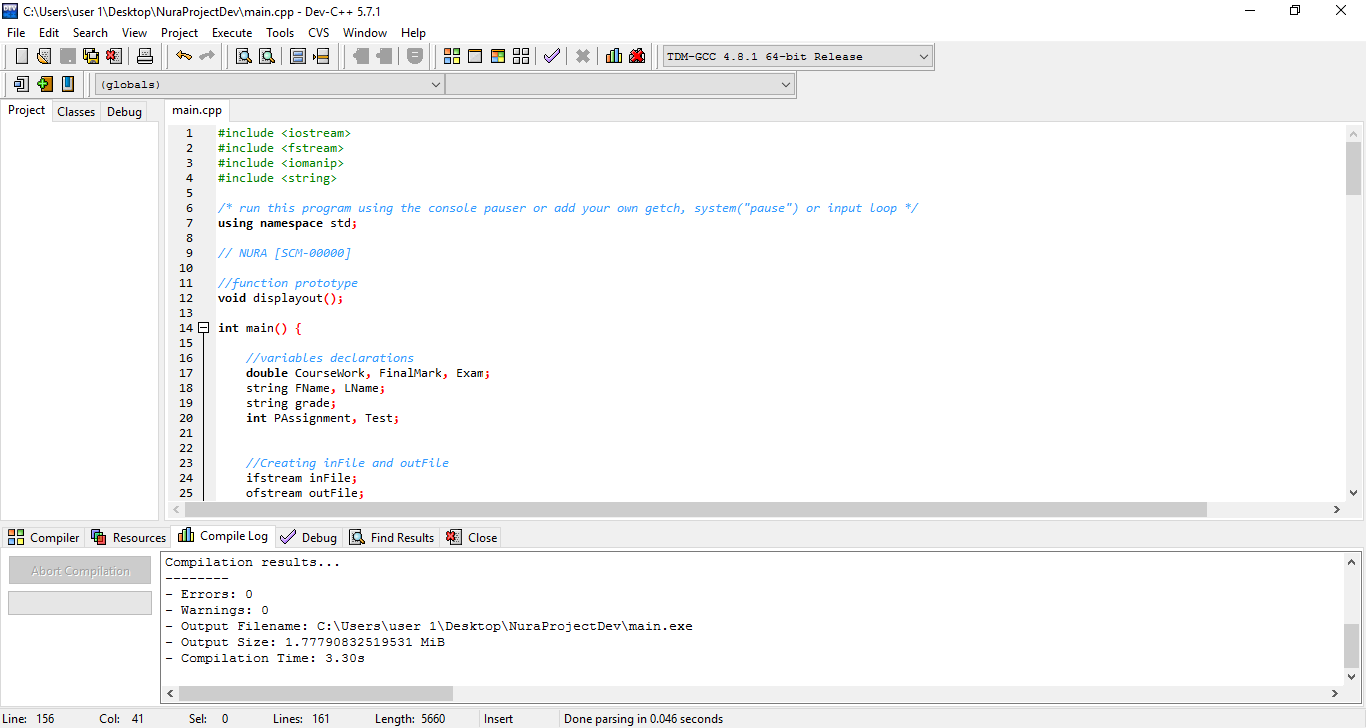
*cout<<" Average Mark : " << " 59.02 " <<endl;*

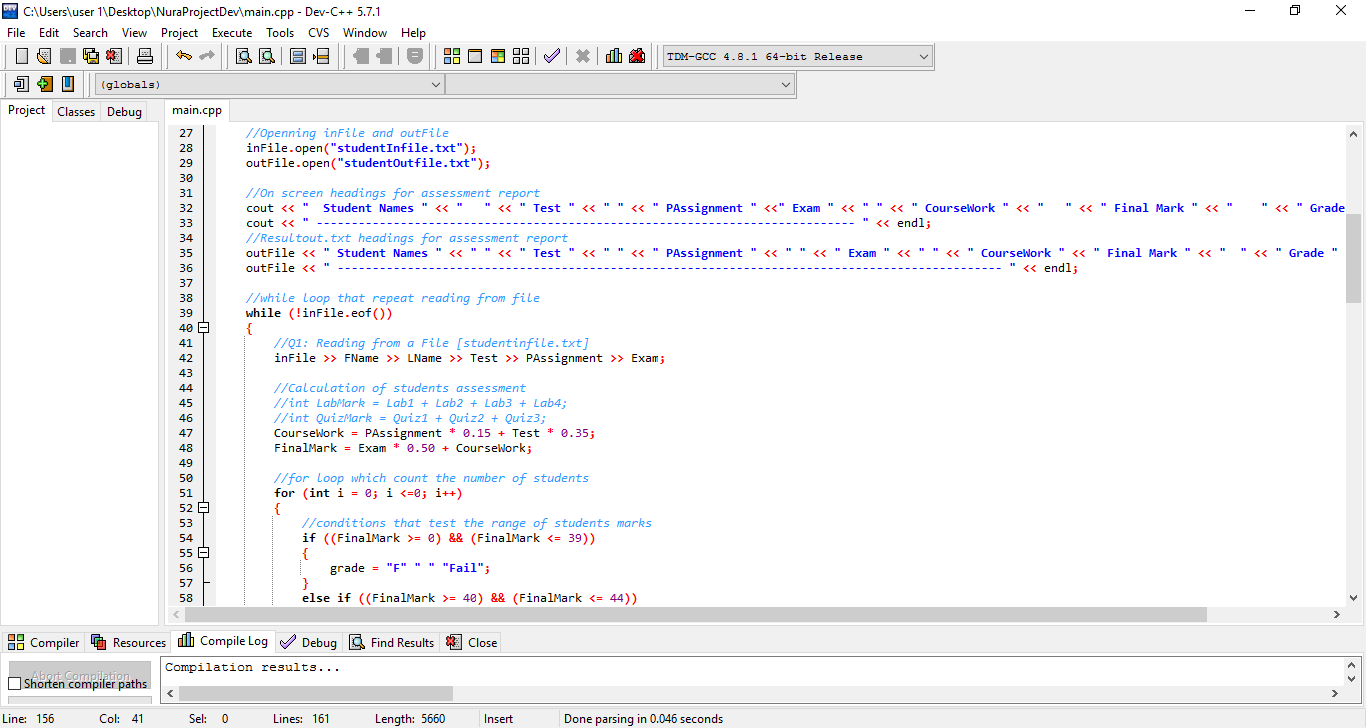
*cout<<" Highest Mark : " << " 81.58 " <<endl;*

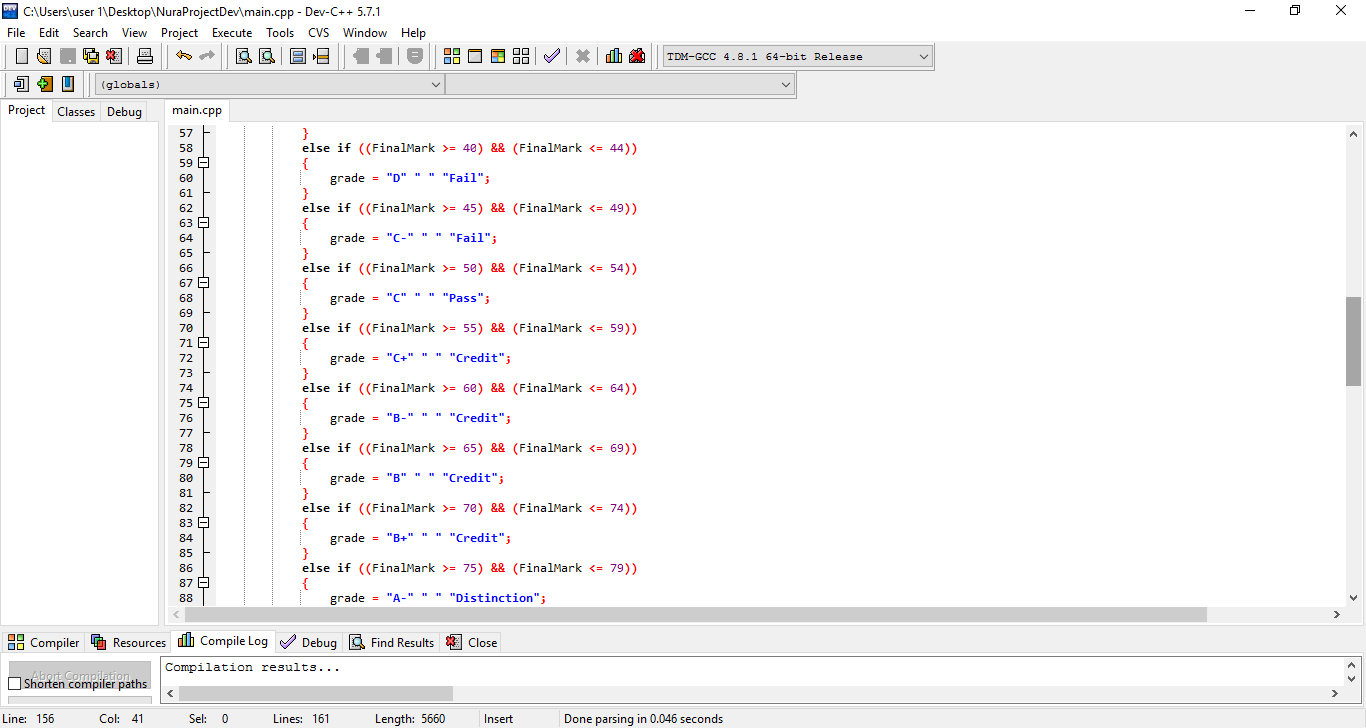
*cout<<" Lowest Mark : " << " 35.36 " <<endl;*

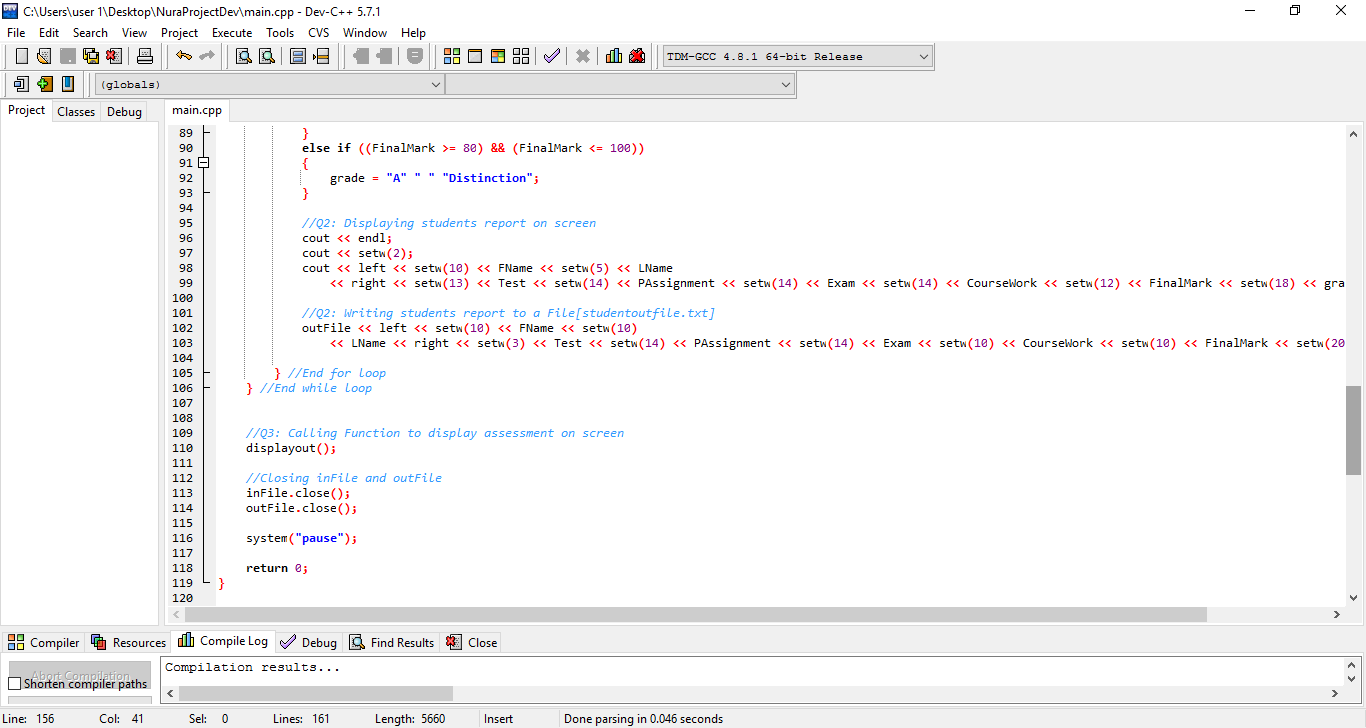
*} //End Function*

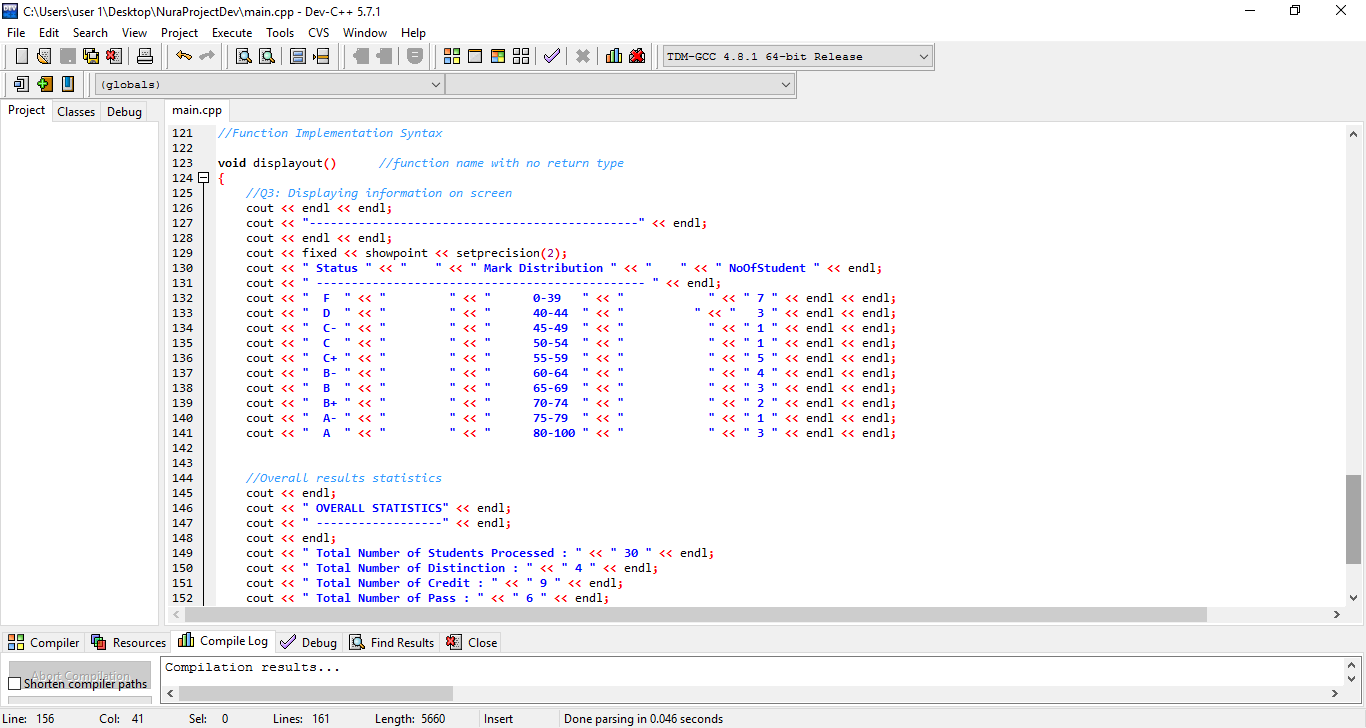
Source Code Screenshots







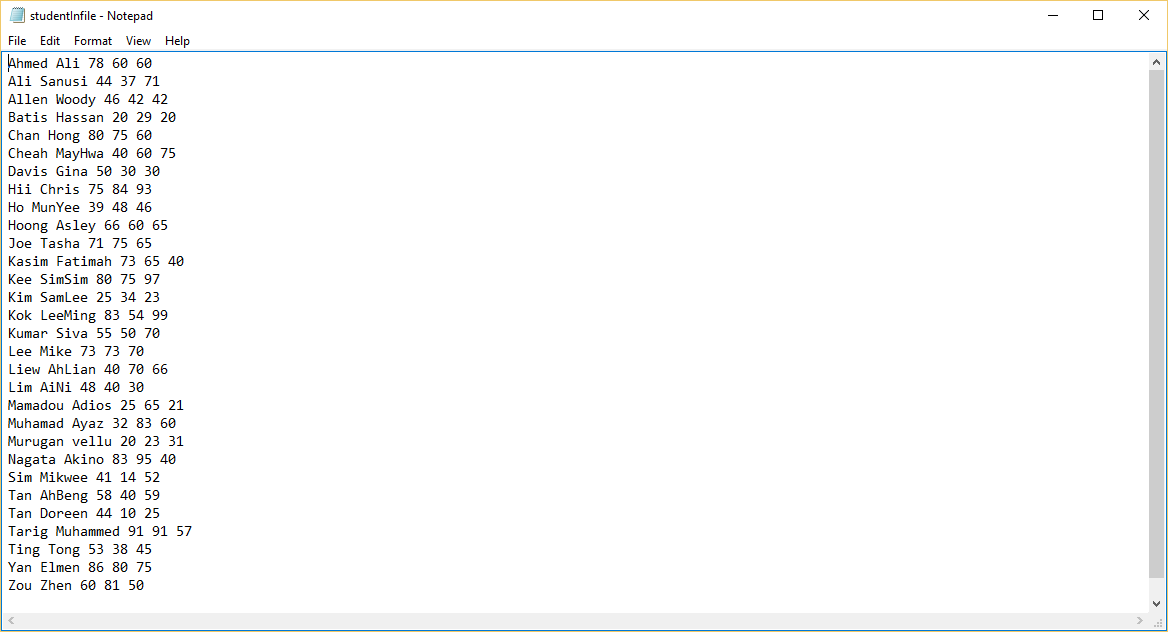




**THE CONTENT OF INPUT FILE AND OUTPUT FILE**

*Section C*

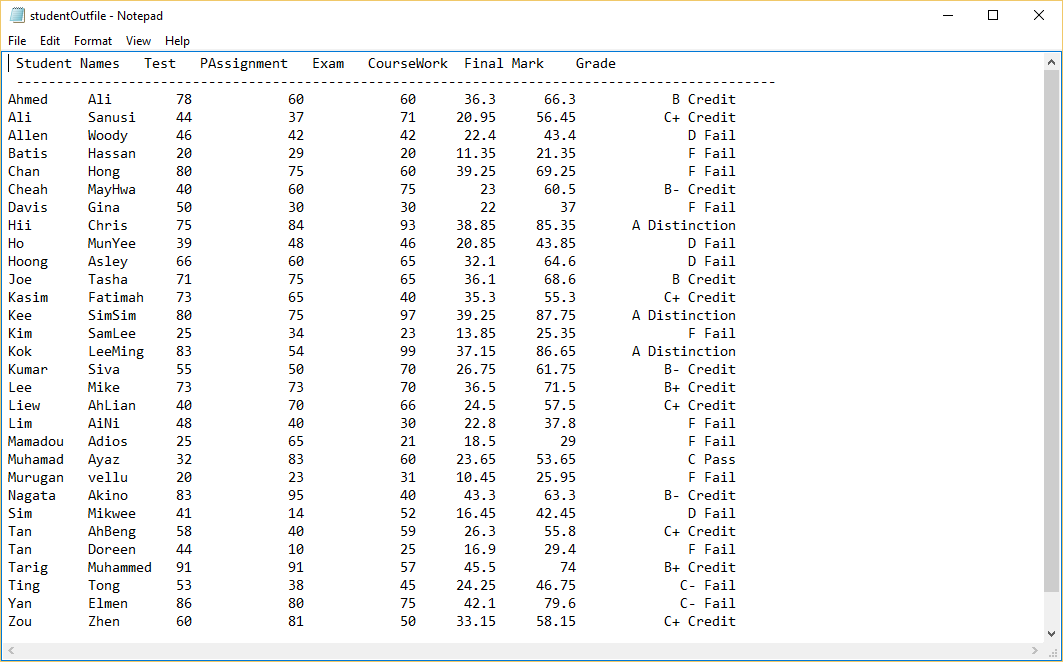
* *CONTENTS OF INPUT FILE*



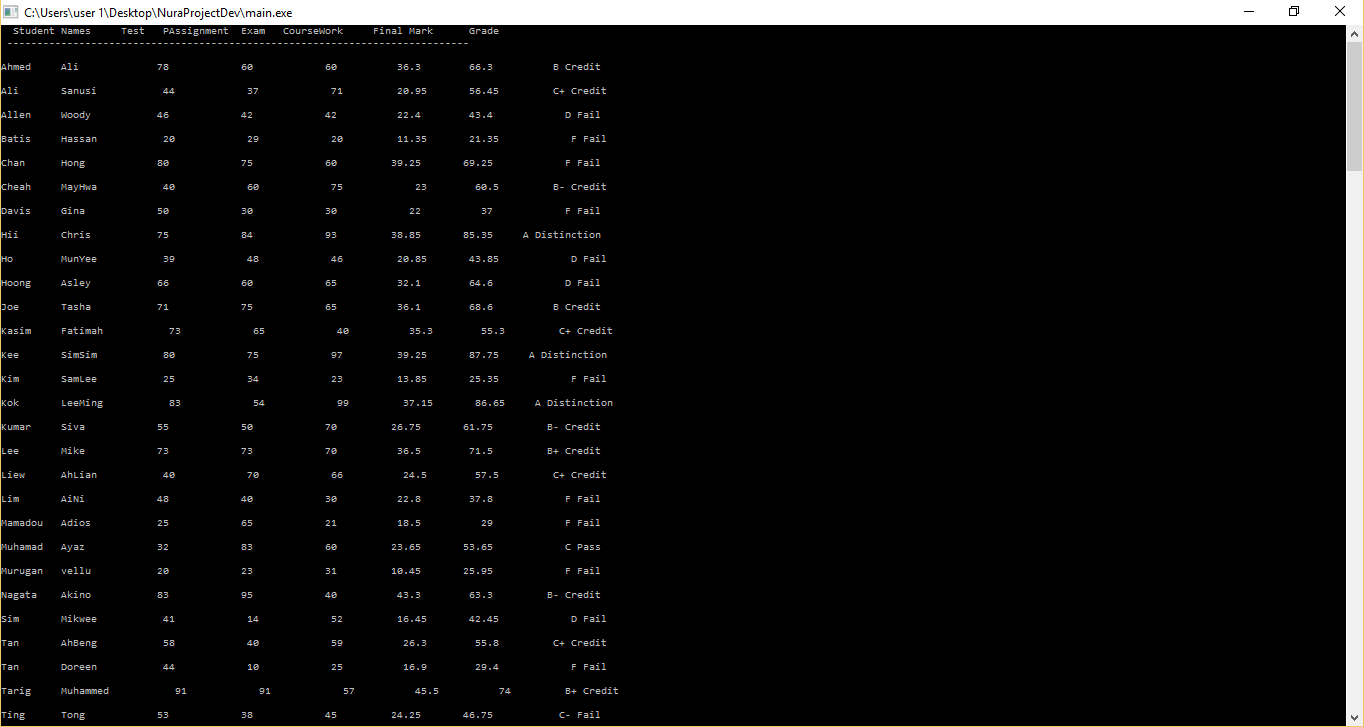
(***SECOND PART OF SECTION C****)*

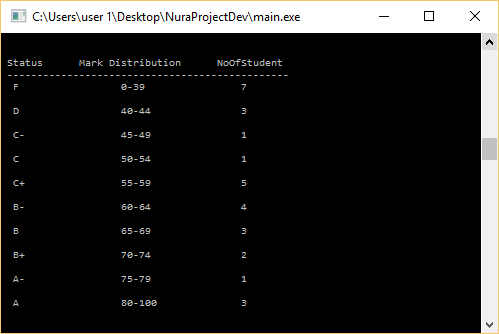
* *CONTENTS OF OUTPUT FILE*

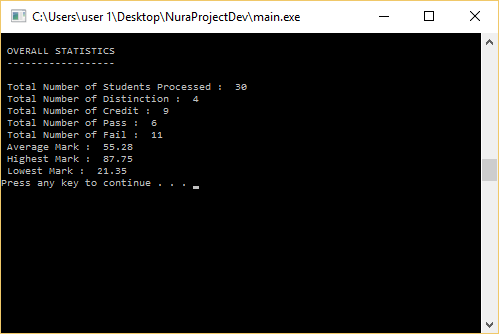
**FINAL OUTPUT**



OUTPUT SCREEN







**ASSUMPTION**

*This Assignment has been concluded by me, the program is built to read a file and perform some arithmetic’s calculations which would provide the result and store it in a new file. I am obviously sure the program has been completed successfully without any inconvenient. For any errors that might arise due to an unexpected threat, I would be available to respond to any kind of queries concerning the program.*

***Developed By:***

*NURA*

**REFERENCE**

|  |  |  |  |
| --- | --- | --- | --- |
| ***Author*** | ***Book Title*** | ***Edition*** | ***Year Published*** |
|  |  |  |  |
|  |  |  |  |