```
/*****************************************************************************
*  AUTHOR        :  Blake Allard
*  STUDENT ID    :  358888
*  ASSIGNMENT #5 :  Selection / Repetition: GPA Calculator
*  CLASS         :  CS1A
*  SECTION       :  M/W: 8am
*  DUE DATE      :  11/06/24
*****************************************************************************/

#include <iostream>       /* cout, cin           */
#include <iomanip>        /* setprecision, fixed */
using namespace std;

/*****************************************************************************
* INPUT & OUTPUT: GPA Calculator
*_____________________________________________________________________________
* This program will obtain:
*        - a student's gpa per entry
*
*
* Then calculate, store, and output:
*        - the total amount of grade points
*        - the average gpa of all entered
*_____________________________________________________________________________
* INPUT:
*                 letterGrade       : The student's gpa
*
* OUTPUT: This program will output:
*                 totalGradePoints  : The total grade points according to each
letter
*                                     grade entered
*           gpa               : The average of all gpa entered
*
*----------------------------------------------------------------------------
*EXAMPLE INPUT / OUTPUT:
*
TEST CASE # 1:
* Enter Letter Grade #1: A
* Enter Letter Grade #2: b
* Enter Letter Grade #3: C
* Enter Letter Grade #4: X
*
*Total Grade Points: 9
*GPA: 3.00
*
*****************************************************************************/


int main()
{

      /*************************************************************************
     * CONSTANTS
       *-------------------------------------------------------------------------
       * OUTPUT - Used for Class Headings
       * ------------------------------------------------------------------------
       * PROGRAMMER         : Programmer's Name
       * CLASS              : Student's Course
       * SECTION            : Class Days and Times
```

```
 * ASN_NUM            : Assignment Number
 * ASN_NAME           : Title of the Assignment
 ****************************************************************************/

// OUTPUT - USED FOR CLASS HEADING
const char PROGRAMMER[]      = "Blake Allard";
const char CLASS[]           = "CS1A";
const char SECTION[]         = "M/W: 8am";
const int  ASN_NUM           = 5;
const char ASN_NAME[]        = "Selection & Repetition: GPA Calculator";

//CONSTANTS - set number of test runs required for program
const short MAX_TESTS = 3;

//CONSTANTS - switch case constant values
const short GPA_A = 4;
const short GPA_B = 3;
const short GPA_C = 2;
const short GPA_D = 1;
const short GPA_F = 0;

/*****************************************************************************
 * INITIALIZATIONS -
 ****************************************************************************/
int    totalGrades      = 0;
int    letterGradeCount = 0;
int    gradePoints      = 0;


/*****************************************************************************
 * VARIABLES
 ****************************************************************************/
char   letterGrade;        // IN   & CALC - obtained from user input and
                           //               processes in-loop calculations

char   invalidLetterGrade; // IN   & CALC - obtained from user if input is
                           //               invalid and error checks input

double gpa;                // CALC & OUT  - calculates and outputs gpa in
                           //               and out of the loop

int    testCase;           // CALC & OUT  - testCase is calculated to run
                           //               a set number of loop iterations
                           //               and outputs testCase # each run
/*****************************************************************************
 * OUTPUT - Class heading
 ****************************************************************************/

cout <<  left;
cout <<  "*****************************************************\n";
cout <<  "*  PROGRAMMED BY :  "        << PROGRAMMER
   << endl;
cout <<  "*  "             << setw(14) << "CLASS"   << ":  " << CLASS
      << endl;
cout <<  "*  "             << setw(14) << "SECTION" << ":  " << SECTION
      << endl;
cout <<  "*  ASSIGNMENT #" << setw(2)  <<  ASN_NUM  << ":  " << ASN_NAME
      << endl;
cout <<  "*****************************************************\n\n";
```

```
      cout << right;



//FOR - loop output requires 3 runs of testing
for (testCase = 1; testCase <= MAX_TESTS; testCase ++)
{

      letterGradeCount = 0;
      totalGrades      = 0;




      /*************************************************************************
       * INPUT - prompt the user to provide:
       *                                        -letterGrade
       *************************************************************************/

      // OUTPUT - prompt
      cout << "TEST CASE # " << testCase << endl;

      //VALIDATION - validate the user's letter grade input as a defined value
      do
      {


      // INPUT - take letterGrade input from the user
      cout << "\tEnter Letter Grade #" << letterGradeCount + 1 << ": ";
      cin.get(letterGrade);
      letterGrade = toupper(letterGrade);
      cin.ignore(100, '\n');


      //INITIALIZE - define invalidLetterGrade to validate valid letter grades
      invalidLetterGrade = (letterGrade != 'A' &&
                            letterGrade != 'B' &&
                            letterGrade != 'C' &&
                            letterGrade != 'D' &&
                            letterGrade != 'F' &&
                            letterGrade != 'X');

      //OUTPUT - error check user's input when letterGrade value is not a valid
      //         input

            if (invalidLetterGrade)
            {
                  cout << "Invalid grade!  Please input a valid letter grade: \n\
n";
            }



      }while (invalidLetterGrade); //... end do loop


      /*************************************************************************
       * PROCESSING - when letterGrade is not 'X' then calculate:
       *                               - totalGrades = +totalGrades + gradePoints
```

```
 *                              - letterGradeCount = letterGradeCount + 1
 ***************************************************************************/

      while (letterGrade != 'X')
      {


                  //PROCESSING - pair user character input with matching processing
                  switch(letterGrade)
                  {
                            case     'A'   : gradePoints = GPA_A;
                                             break;

                            case     'B'   : gradePoints = GPA_B;
                                             break;

                            case     'C'   : gradePoints = GPA_C;
                                             break;

                            case     'D'   : gradePoints = GPA_D;
                                             break;

                            case     'F'   : gradePoints = GPA_F;
                                             break;


                  }  //... end switch


                  // PROCESSING - if the user inputs a character with a defined
value
                  //              held in the switch statement , the value will be
                  //              added to totalGrades and letterGraeCount will be
                  //              incremented


                  totalGrades     += gradePoints;
                  letterGradeCount = letterGradeCount + 1;

      do
      {

            // INPUT & VALIDATE - take letterGrade input from the user
            cout << "\tEnter Letter Grade #" << letterGradeCount + 1 << ": ";
            cin.get(letterGrade);
            letterGrade = toupper(letterGrade);
            cin.ignore(100, '\n');

            invalidLetterGrade = (letterGrade != 'A' &&
                                  letterGrade != 'B' &&
                                  letterGrade != 'C' &&
                                  letterGrade != 'D' &&
                                  letterGrade != 'F' &&
                                  letterGrade != 'X');
         //OUTPUT - error check user's input when letterGrade value is not a valid
         //         input
            if (invalidLetterGrade)
            {
                  cout << "Invalid grade!  Please input a valid letter grade: \n\
```

```
n";
            }


      }while (invalidLetterGrade); //... end do loop


      } //... end while (letterGrade != 'X')

    /*************************************************************************
       * OUTPUT - when letterGrade equals 'X' output:
       *                          - gpa
       *                          - totalGrades
       *************************************************************************/

      //OUTPUT - output based on the check and change of for loop count)
      if (letterGradeCount != 0)
      {

            //CALCULATING - running total of grade letter inputs / number of inputs
            gpa = double(totalGrades) / letterGradeCount;

            //FORMATTING - set correct precision and fixed values for gpa OUPUT
            cout << setprecision(2) << fixed << endl;

            //OUTPUT - output the total grade points & the GPA average
            cout << "Total Grade Points: " << totalGrades << endl;
            cout << "GPA: "                << gpa         << endl << endl << endl;

            //FORMATTING - reset setprecision and fixed back to default
            cout << setprecision(6);
            cout.unsetf(ios::fixed);

      } //..end if

}   //... end for loop

      return 0;
}
```