LOOPING PROBLEM WORKSHEET

(indicate if the input is the LCV if it isn't does it go in or out of the loop)

INPUT: gradeLetter (LCV)

(indicate if the output goes in the loop or out of the loop)

OUTPUT: gpa (out of loop)
gradePts (out of loop)

PROCESSING

IN LOOP: classCount += 1
gradeNum += 1

OUT OF LOOP: gpa = float(gradePts)/classCount

INITIALIZATIONS: classCount = 0
gradNum = 1

FOR LOOP:

How many times will it run? [ ]

WHILE LOOP:

LCV: gradeLetter
Sentinel Value: 'X'
Condition: gradeLetter != 'X'

BASIC FORMATS

Loops processing user input

INITIALIZATIONS

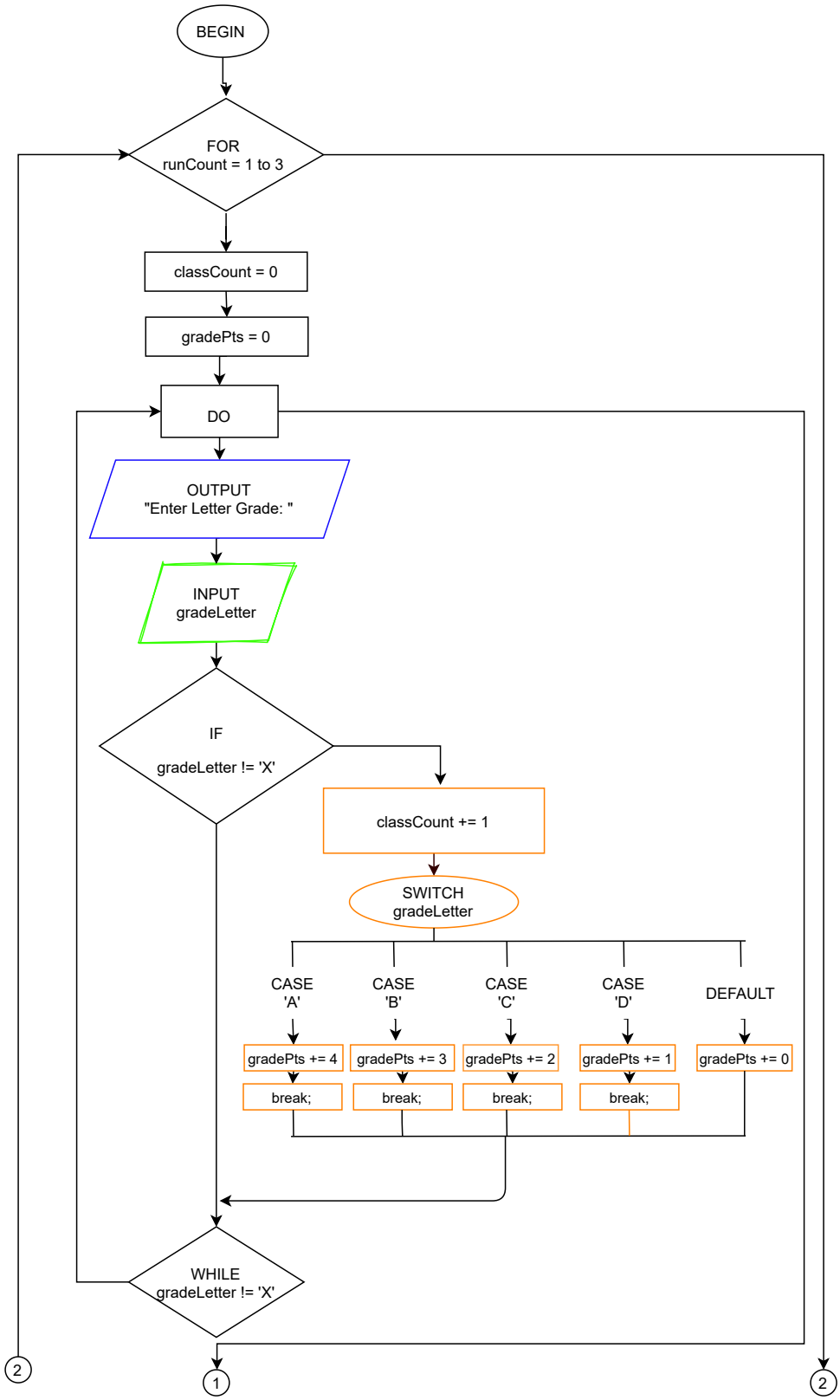
FOR count = 1 to *X*
    INPUT
    PROCESSING (IN LOOP)
    OUTPUT (IN LOOP)
END FOR

PROCESSING (OUT OF LOOP)
OUTPUT (OUT OF LOOP)

INITIALIZATIONS

INPUT LCV
WHILE *Condition*
    PROCESSING (IN LOOP)
    OUTPUT (IN LOOP)
    INPUT LCV
END WHILE

PROCESSING (OUT OF LOOP)
OUTPUT (OUT OF LOOP)

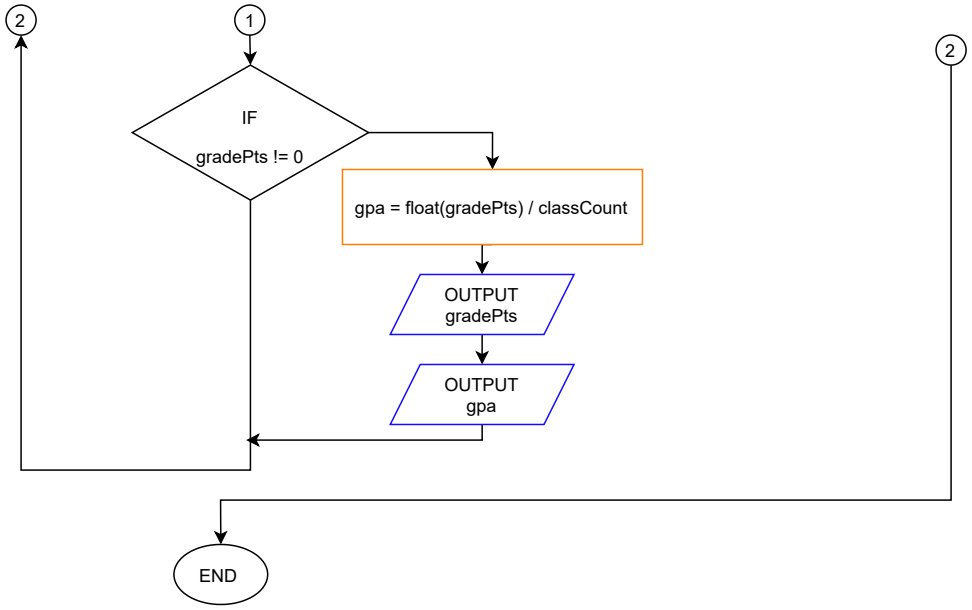


Variable List

INPUT
gradeLetter

PROCESSING
classCount
runCount

OUTPUT
gradePts
gpa


2
1
IF
gradePts != 0
gpa = float(gradePts) / classCount
OUTPUT
gradePts
OUTPUT
gpa
2
END

```
 1 ************************************************
 2 * PROGRAMMED BY : Andrew Gharios
 3 * CLASS         : CS1A
 4 * SECTION       : MW 8:00a - 10:30a
 5 * ASSIGMENT #5  : Selection & Repetition
 6 ************************************************
 7
 8 TEST CASE # 1:
 9     Enter Letter Grade #1: A
10     Enter Letter Grade #2: b
11     Enter Letter Grade #3: C
12     Enter Letter Grade #4: X
13
14 Total Grade Points: 9
15 GPA: 3.00
16
17
18 TEST CASE # 2:
19     Enter Letter Grade #1: a
20     Enter Letter Grade #2: b
21     Enter Letter Grade #3: A
22     Enter Letter Grade #4: d
23     Enter Letter Grade #5: C
24     Enter Letter Grade #6: F
25     Enter Letter Grade #7: a
26     Enter Letter Grade #8: x
27
28 Total Grade Points: 18
29 GPA: 2.57
30
31
32 TEST CASE # 3:
33     Enter Letter Grade #1: X
34
```

```cpp
/*****************************************************
* AUTHOR         : Andrew Gharios
* STUDENT ID     : 1449366
* ASSIGNMENT #5 : Selection & Repetition
* CLASS          : CS1A
* SECTION        : MW 8AM-10:20AM
* DUE DATE       : 4/19/21
*****************************************************/

#include <iostream>
#include <iomanip>
#include <cstring>
using namespace std;

/*****************************************************************************
* REPETITION
*----------------------------------------------------------------------------
* This program will take in a letter Grade and calculate total points and GPA
* earned based on the grade. User can input as many grades as they want and can
* exit the program by inputting an X.
*----------------------------------------------------------------------------
* INPUT:
* The user will input the grade letter they wish.
*
* OUTPUT:
* This program will output the total points based on letters inputted and GPA.
*****************************************************************************/

int main()
{
    /*************************************************************************
     * CONSTANTS
     * -----------------------------------------------------------------------
     * OUTPUT - USED FOR CLASS HEADING
     * -----------------------------------------------------------------------
     * PROGRAMMER     : Programmer's Name
     * CLASS          : Student's Course
     * SECTION        : Class Days and Times
     * ASSIGNMENT #4  : Assignment's name.
     * -----------------------------------------------------------------------
     * OUTPUT - USED FOR PROCESSING
     * -----------------------------------------------------------------------
     * END_LOOP   : Indicates how many sets of sheep ages there will be
     *************************************************************************/
    const char PROGRAMMER[]      = "Andrew Gharios";
    const char CLASS[]           = "CS1A";
    const char SECTION[]         = "MW 8:00a - 10:30a";
    const char AS_NAME[]         = "Selection & Repetition";

    const int  END_LOOP     = 3;


    char    gradeLetter;   // IN & CALC  - Grade letter input.
    int     runCount;      // CALC       - LCV for for loop.
    int     classCount;    // CALC       - Counts how many classes inputted.
    int     gradeNum;      // CALC       - total age of all sheep in
    int     gradePts;      // CALC & OUT - Grade points accumulated from grades.
    float   gpa;           // CALC & OUT - total GPA of user based on inputs.

    /*************************************************************************
     * OUTPUT - class heading
     *************************************************************************/
```

```
63      cout << left;
64      cout << "*************************************************\n";
65      cout << "* PROGRAMMED BY : "  << PROGRAMMER << endl;
66      cout << "* "      << setw(14) << "CLASS"    << ": " << CLASS    << endl;
67      cout << "* "      << setw(14) << "SECTION"  << ": " << SECTION  << endl;
68      cout << "* "      << setw(14) << "ASSIGMENT #5" << ": " << AS_NAME << endl;
69      cout << "*************************************************\n\n";
70      cout << right;
71
72      /*****************************************************************************
73       * INPUT - user inputs their grade letter, program makes sure the letters
74       * are all capitalized.
75       *****************************************************************************/
76      for (runCount = 1; runCount <= END_LOOP; runCount = runCount + 1)
77      {
78          gradeNum   = 1;
79          classCount = 0;
80          gradePts   = 0;
81
82          cout << left;
83          cout << "TEST CASE # " << runCount << ":" << endl;
84
85          do
86          {
87              // Getting letter grade input from user and making sure its capital.
88              cout << "\tEnter Letter Grade #" << gradeNum << ": ";
89              gradeNum   += 1;
90              cin.get(gradeLetter);
91              cin.ignore(10000, '\n');
92              gradeLetter = toupper(gradeLetter); // making input is uppercase.
93
94      /*****************************************************************************
95       * PROCESSING - the program checks how many points are given based on the
96       * letter grades, and then calculate the GPA.
97       *****************************************************************************/
98
99
100             // Calculating GPA as long as gradeLetter inputed is not X.
101             if(gradeLetter != 'X')
102             {
103                 classCount += 1; // classcounter for GPA calculation.
104                 switch(gradeLetter)
105                 {
106                 case 'A' : gradePts += 4;
107                            break;
108                 case 'B' : gradePts += 3;
109                            break;
110                 case 'C' : gradePts += 2;
111                            break;
112                 case 'D' : gradePts += 1;
113                            break;
114                 default  : gradePts += 0;
115                            break;
116                 }
117             }
118         }while(gradeLetter != 'X');
119
120         if(gradePts != 0)
121         {
122             gpa = float(gradePts) / classCount;
123
124     /*****************************************************************************
```

```
125      * OUTPUT - The program outputs how many points are accumulated as well as
126      *          the total GPA according to grades inputted.
127      *************************************************************************/
128
129             cout << fixed;
130             cout << setprecision(2);
131             cout << endl;
132             cout << "Total Grade Points: " << gradePts << endl;
133             cout << "GPA: " << gpa << endl << endl     << endl;
134          }
135     }
136
137     return 0;
138
139 }
140
```