

Programming with C++

COMP2011: Examples on C++ Basics and Controls

Cecia Chan

Gary Chan

Cindy Li

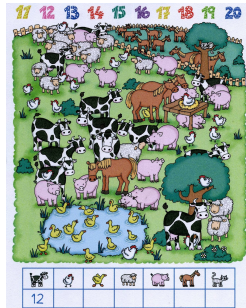
Wilfred Ng

Department of Computer Science & Engineering
The Hong Kong University of Science and Technology
Hong Kong SAR, China



Part I

Count Animals



The Count Animals Problem

- There are two types of animals, pigs and sheeps in a farm.
- Each pig weighs 4.5 units and each sheep weighs 3 units.
- The total weight of animals in a barn should be exactly 36 units.
- List out all possible combinations of pigs and sheeps in the farm.

Solution:

$$0 * 4.5 + 12 * 3 = 36$$

$$2 * 4.5 + 9 * 3 = 36$$

$$4 * 4.5 + 6 * 3 = 36$$

$$6 * 4.5 + 3 * 3 = 36$$

$$8 * 4.5 + 0 * 3 = 36$$

First Attempt: What's Wrong?

```
1  #include <iostream>          /* File: two-animals-v1.cpp */
2  using namespace std;
3
4  int main()
5  {
6      float wt_pig = 4.5;
7      float wt_sheep = 3.0;
8      float total_wt = 36;
9
10     for (int num_pigs = 0; num_pigs * wt_pig <= total_wt;
11         ↪ num_pigs++)
12     {
13         float remain_wt = total_wt - num_pigs * wt_pig;
14         int num_sheeps = remain_wt / wt_sheep;
15         remain_wt = remain_wt % wt_sheep;
16
17         if (remain_wt == 0)
18             cout << num_pigs << " * " << wt_pig << " + "
19                 << num_sheeps << " * " << wt_sheep << " = "
20                 << total_wt << endl;
21     }
22     return 0;
23 }
```

Second Attempt: Any Problem?

```
1  #include <iostream>          /* File: two-animals-v2.cpp */
2  using namespace std;
3
4  int main()
5  {
6      float wt_pig = 4.5;
7      float wt_sheep = 3.0;
8      float total_wt = 36;
9
10     for (int num_pigs = 0; num_pigs * wt_pig <= total_wt;
11         ↪ num_pigs++)
12     {
13         float remain_wt = total_wt - num_pigs * wt_pig;
14         int num_sheeps = remain_wt / wt_sheep;
15         remain_wt -= num_sheeps * wt_sheep;
16
17         if (remain_wt == 0)
18             cout << num_pigs << " * " << wt_pig << " + "
19                 << num_sheeps << " * " << wt_sheep << " = "
20                 << total_wt << endl;
21     }
22     return 0;
23 }
```

Problems of Comparing Floating-point Numbers

```
1  #include <iostream>          /* File: float-comparison-v1.cpp */
2  using namespace std;
3
4  int main()
5  {
6      float x = 0.1;
7      float product = 10.0 * x;
8
9      float sum = 0.0;
10     for (int i = 0; i < 10; ++i)
11         sum += x;
12
13     cout << "sum = " << sum << endl;
14     cout << "product = " << product << endl;
15     cout << "10.0 * x = " << 10.0 * x << endl;
16     cout << (sum == product) << endl;
17
18     return 0;
19 }
```

Problems of Comparing Floating-point Numbers ..

```
1  #include <iostream>          /* File: float-comparison.cpp */
2  using namespace std;
3
4  int main()
5  {
6      float x = 0.1;
7      float product = 10.0 * x;
8
9      float sum = 0.0;
10     for (int i = 0; i < 10; ++i)
11         sum += x;
12
13     // Set output precision to 10 significant figures
14     cout.precision(10);
15     // Print boolean outputs as true or false instead of 1 and 0
16     cout << boolalpha;
17
18     cout << "sum = " << sum << endl;
19     cout << "product = " << product << endl;
20     cout << "10.0 * x = " << 10.0 * x << endl;
21     cout << (sum == product) << endl;
22     return 0;
23 }
```

Count Animals: Further Work

- Further check ways to compare floating point numbers [here](#).
- What if the total number of animals is not more than 10.
- What if we have three types of animal instead of two? (You may ignore the constraint on the total number of animals.)
- For those who know recursion, can you work out a recursive solution?

Part II

GPA Calculator



- Assume the following letter grade to grade point conversion:

Letter Grade	Grade Point
A	4.0
B	3.0
C	2.0
D	1.0
F	0.0

- Design a program that calculates a student's GPA (grade point average).

Typical Output

```
No.  of credits of your course (0 to stop):  3
Your letter grade (A, B, C, D or F):  A
No.  of credits of your course (0 to stop):  4
Your letter grade (A, B, C, D or F):  B
No.  of credits of your course (0 to stop):  2
Your letter grade (A, B, C, D or F):  E
Invalid input, please enter your grade again!
No.  of credits of your course (0 to stop):  2
Your letter grade (A, B, C, D or F):  D
No.  of credits of your course (0 to stop):  0
You have taken a total of 9 credits ...
and your GPA is 2.88889
```

Program Requirements

- A student first enters the number of credits of his/her course.
- The program stops if the number of credits is ≤ 0 .
- The student then enters the letter grade A, B, C, D or F.
- Invalid letter grades are ignored and the student is prompted to re-enter the grade.
- The program shall calculate the total number of credits earned by the student and his/her GPA according to the following formula:

$$\frac{\sum_{i=1}^n (grade_i * credit_i)}{\sum_{i=1}^n credit_i}$$

Major components of the program:

- ① A loop for each taken course to
 - Ⓐ ask for the number of credits
 - Ⓑ ask for the letter grade
 - Ⓒ convert the letter grade to grade points
 - Ⓓ accumulate the total credits and grade points
- ② Calculate and output the GPA.

You will need some variables to hold:

- the number of credits
- the letter grade
- the converted grade points
- the sum of credits
- the sum of grade points

Variables

```
1 char grade;
2 /* Don't forget to initialize some of these variables */
3 int num_credits, total_num_credits = 0;
4 int total_grade_points = 0;
5 ↪ // Is it a good idea to use integer type here?
6 // Think about the output
7 cout << "You have taken a total of " << total_num_credits
8     << " credits ..." << endl << "and your GPA is "
9     << total_grade_points / total_num_credits << endl;
```

Recall the **usual arithmetic conversion** rules for binary operations:

- If all operands are int
 - compute using integer arithmetic
- If one operand is double/float
 - convert the other operand to double/float
 - compute using floating-point arithmetic
 - return the result in double/float

GPA Calculator: Using `if`

```
1  char grade;
2  int num_credits, total_num_credits = 0;
3  double total_grade_points = 0;
4
5  cout << "No. of credits of your course (0 to stop): ";
6  cin >> num_credits;
7  cout << "Your letter grade (A, B, C, D or F): ";
8  cin >> grade;
9  total_num_credits += num_credits; // Update total no. of credits
10
11 if (grade == 'A')           // Convert letter grade to grade point
12     total_grade_points += num_credits * 4;
13 else if (grade == 'B')
14     total_grade_points += num_credits * 3;
15 else if (grade == 'C')
16     total_grade_points += num_credits * 2;
17 else if (grade == 'D')
18     total_grade_points += num_credits * 1;
19 else if (grade == 'F')
20     total_grade_points += num_credits * 0;
21 else
22     total_num_credits -= num_credits;
```

GPA Calculator: Using **switch**

```
1  /*
2   * Codes for variables definition and initialization, and inputs
3   */
4
5  switch (grade)           // Convert letter grade to grade point
6  {
7      case 'A':
8          total_grade_points += num_credits * 4; break;
9      case 'B':
10         total_grade_points += num_credits * 3; break;
11     case 'C':
12         total_grade_points += num_credits * 2; break;
13     case 'D':
14         total_grade_points += num_credits * 1; break;
15     case 'F':
16         total_grade_points += num_credits * 0; break;
17     default:
18         total_num_credits -= num_credits;
19 }
```


GPA Calculator: Using `if` .. Allowing Small Case

```
1  /*
2   * Codes for variables definition and initialization, and inputs
3   */
4
5  if ((grade == 'A') || (grade == 'a'))
6      total_grade_points += num_credits * 4;
7
8  else if ((grade == 'B') || (grade == 'b'))
9      total_grade_points += num_credits * 3;
10
11 else if ((grade == 'C') || (grade == 'c'))
12     total_grade_points += num_credits * 2;
13
14 else if ((grade == 'D') || (grade == 'd'))
15     total_grade_points += num_credits * 1;
16
17 else if ((grade == 'F') || (grade == 'f'))
18     total_grade_points += num_credits * 0;
19
20 else
21     total_num_credits -= num_credits;
```

GPA Calculator: Using `switch` .. Allowing Small Case

```
1  switch (grade)                // Convert letter grade to grade point
2  {
3      case 'A':
4      case 'a':
5          total_grade_points += num_credits * 4; break;
6      case 'B':
7      case 'b':
8          total_grade_points += num_credits * 3; break;
9      case 'C':
10     case 'c':
11         total_grade_points += num_credits * 2; break;
12     case 'D':
13     case 'd':
14         total_grade_points += num_credits * 1; break;
15     case 'F':
16     case 'f':
17         total_grade_points += num_credits * 0; break;
18     default:
19         total_num_credits -= num_credits;
20 }
```

GPA Calculator: Complete Program .. Using constants

```
1  #include <iostream>      /* File: gpa.cpp */
2  using namespace std;
3
4  const double A = 4.0;    // Definition of Constants
5  const double B = 3.0;
6  const double C = 2.0;
7  const double D = 1.0;
8  const double F = 0.0;
9
10 int main()
11 { // Variables definition and initialization
12     char grade;
13     int num_credits, total_num_credits = 0;
14     double total_grade_points = 0;
15
16     do
17     {
18         cout << "No. of credits of your course (0 to stop): ";
19         cin >> num_credits;
20
21         if (num_credits <= 0) // What does this do?
22             break;
23
24         cout << "Your letter grade (A, B, C, D or F): ";
25         cin >> grade;
26         total_num_credits += num_credits; // Update total no. of credits
27     }
```

GPA Calculator: Complete Program .. Using constants ..

```
28 switch (grade) // Convert letter grade to grade point
29 {
30     case 'A': // No break here; execute code in case 'a'
31     case 'a':
32         total_grade_points += num_credits * A; break;
33     case 'B':
34     case 'b':
35         total_grade_points += num_credits * B; break;
36     case 'C':
37     case 'c':
38         total_grade_points += num_credits * C; break;
39     case 'D':
40     case 'd':
41         total_grade_points += num_credits * D; break;
42     case 'F':
43     case 'f':
44         total_grade_points += num_credits * F; break;
45     default :
46         cout <<
47             ↳ "Invalid input, please enter your grade again!\n";
48         total_num_credits -= num_credits;
49 } while (true); // Why is this not an infinite loop?
```

GPA Calculator: Complete Program .. Using constants ..

```
53
54     cout << "You have taken a total of " << total_num_credits
55           << " credits ..." << endl << "and your GPA is "
56           << total_grade_points / total_num_credits << endl;
57
58     return 0;
59 }
```