Homework 5

Dr. Manna

CS 10 | 30 points | due: 02/15/17 @ 11:59 pm

Problem statement

1. Using the code snippet provided, please answer the following questions:

```
001  const int LIMIT = 50;
002  int AddEm(int x, int y);
003  int main() {
004   int x = 42,
005   y = 35;
006   int Sum;
007   Sum = AddEm(x, y);
008   return 0;
009  }
010  int AddEm(int x, int y) {
011   int Total;
012   Total = x + y;
013   if (Total > LIMIT)
014   Total = 0;
015   return (Total);
016  }
```

   a. (1 point) What is the scope of the identifier `Sum` that is declared in `Line 6`?
      i. Line 1 to Line 16
      ii. Line 6 to Line 16
      iii. Line 6
      iv. Line 6 to Line 7
      v. Line 6 to Line 9
      vi. None of these

   b. (1 point) What is the scope of the identifier x that is declared in Line 4?
      i. Line 1 to Line 16
      ii. Line 4 to Line 16
      iii. Line 4
      iv. Line 4 to Line 7
      v. Line 4 to Line 9
      vi. None of these

   c. (1 point) What is the scope of the identifier x that is declared in Line 10?

i. Line 1 to Line 16
ii. Line 4 to Line 16
iii. Line 10
iv. Line 10 to Line 12
v. Line 10 to Line 16
vi. None of these

d. (1 point) What is the scope of the identifier LIMIT that is declared in Line 1?
   i. Line 1 to Line 16
   ii. Line 1 to Line 3
   iii. Line 1
   iv. Line 10 to Line 13
   v. Line 10 to Line 16
   vi. None of these

e. (4 points) Which of the following are true/false?
   i. LIMIT is local to main()
   ii. Total is local to AddEm()
   iii. Sum is local to main()
   iv. LIMIT is global
   v. x is global
   vi. All of them are true
   vii. All but 1 are true
   viii. 2 and 3 only

2. (2 points) What is the new value of x?

```
001  #include <iostream>
002  using namespace std;
003
004  void fun(int& x) {
005    x = 20;
006  }
007  int main() {
008    int x = 10;
009    fun(x);
010    cout << "New value of x is " << x;
011    return 0;
012  }
```

3. (2 points) What is the output of this program?

```
001  #include <iostream>
002  using namespace std;
```

```
003  void Sum(int a, int b, int& c) {
004    a = b + c;
005    b = a + c;
006    c = a + b;
007  }
008  int main() {
009    int x = 2, y = 3;
010    Sum(x, y, y);
011    cout << x << " " << y;
012    return 0;
013  }
```

4. (5 points) What is the output of the following C++ program?

```
#include <iostream>
using namespace std;

int global;

void f1(int);
void f2(int &);

void f1(int n) {
  n += global;
}

void f2(int & n) {
  n += global;
}

int main() {
  cout << global++ << endl;
  {
    int global(100);
    cout << ++global << endl;
  }
  cout << global << endl;
  int local(10);
  cout << local << endl;
  {
    int local(20);
    cout << local + global << endl;
  }
  cout << local + global << endl;
  f1(local);
```

```
  cout << local << endl;
  f1(global);
  cout << global << endl;
  f2(local);
  cout << local << endl;
  f2(global);
  cout << global << endl;
  return 0;
}
```

5. (7 points) The area of an arbitrary triangle can be computed using the formula

   $$Area = \sqrt{s(s-a)(s-b)(s-c)}$$

   where `a`, `b`, and `c` are the lengths of the sides, and `s` is the semiperimeter:

   $$s = (a+b+c)/2$$

   Write a `void` function `compute(...)` that uses five parameters: three value parameters that provide lengths of the edges and two reference parameters that compute the area and perimeter (not the semiperimeter). Make your functions robust. Note that not all combinations of `a`, `b`, and `c` produce a valid triangle. Your function should produce correct results for legal data and reasonable results for illegal combinations.

   [**Hint**: Learn the *Triangle Inequality Theorem*. This theorem simply states that the sum of two sides of a triangle must be greater than the third side. If this is true for all three combinations, then you will have a valid triangle. You'll have to go through these combinations one by one to make sure that the triangle is possible. You can also think of the triangle as having the side lengths a, b, and c and the theorem being an inequality, which states: `a+b > c`, `a+c > b`, and `b+c > a`]

6. (6 points) Write a function named `convertToLowestTerms` that inputs two parameters by reference named `numerator` and `denominator` (assuming them to be int). The function should treat these variables as a fraction and reduce them to lowest terms. For example, if `numerator` is `20` and `denominator` is `60`, then the function should change the variables to `1` and `3`, respectively. This will require finding the greatest common divisor for the numerator and denominator then dividing both variables by that number. If the denominator is zero, the function should return `false`, otherwise the function should return `true`. Write a main that uses `convertToLowestTerms` to test with different values.

Submission instructions

For questions 5.1 to 5.4, please type the outputs in a document and upload a .pdf file and .cpp files for 5.5 and 5.6 respectively to Camino under Assignment→ Homework 5→ …. Please email me or use the discussion board for clarification.