COEN 70: Formal Specification and Advanced Data Structures

Winter 2015

Lab 1: Getting familiar with GNU debugger (gdb)

1. Fix the program so that the object B has the intended string (“Hello World”).

```
/* HelloWorldExample.cpp */
#include <iostream>
#include <string>

using namespace std;

struct object {
      string* s;
};

int main()
{
      object A, B;
      A.s = new string;
      *A.s = "Hello World"; // A has "Hello World"
      B = A; // B has "Hello World"
      *A.s = "Goodbye"; // A has "Goodbye"
      cout << *A.s << endl;
      cout << *B.s << endl;
      return 0;
}
```

2. What is wrong with the following program?

```
/* ArrayExample.cpp */
#include <iostream>
using namespace std;

int main()
{
      int score[10];
      for (int i=1; i <= 10; ++i)
      {
            score[i]=3*i;
            cout << score[i];
      }
return 0;
```

```
}
```

3. Fix the next program so that it computes correctly the surface area and volume of a sphere given a radius.

```
/*SphereExample.cpp*/
#include <iostream>
using namespace std;

const double PI = 3.14159265359;

void GetRadius(double & radius);
void ShowResults(double rad, double area, double vol);

int main()
{
  cout << "Program computes surface area and volume of a sphere.\n";
  double  radius,                          // radius of sphere
          surfaceArea=0,                     // its surface area
          volume=0;                          // its volume

  GetRadius(radius);
  volume = surfaceArea * radius / 3.0;
  surfaceArea = 4.0 * PI * radius * radius;

  ShowResults(radius, surfaceArea, volume);
  return 0;
}

void GetRadius(double& rad)
{
  cout << "Enter radius of sphere: ";
  cin >> rad;
}

void ShowResults(double rad, double area, double vol)
{
  cout << "Radius of sphere is " << rad << " inches\n";
  cout << "Its surface area is " << area << "sq. inches\n"
       << "Its volume is " << vol << " cubic inches.\n\n";
}
```

Steps to follow for all three programs:

1. Create the .cpp files

2. Compile the programs using the -g switch, e.g. g++ -g <file>.cpp -o <file>
3. Invoke gdb.
4. Add to a report the steps you used to debug
5. Fix the problems

Submission

Create the fixed programs and the report and submit as one file using camino. Your username is the same as your Design Center username, and your password is the last seven digits of your ID.