CS 210: Sample Exam #2

Ray S. Babcock

November 7, 2003

Name: ____________________

1. (10 points) Given the listing on the next page

   (a) (6 points) What would print?

   (b) (4 points) Write the code to instantiate two objects called *fog* and *sleet* of the ClassOne class.

```cpp
// one.h CS210 sample F'03
// 11/7/03 - Ray S. Babcock

#ifndef ONE_H_
#define ONE_H_

class ClassOne {
public:
ClassOne();
int getSize();
char getCode();
private:
int size;
char code;
};

class ClassTwo {
public:
ClassTwo();
int getAge();
char getInit();
private:
int age;
char init;
};

ClassOne::ClassOne()
{
  size = 1;
  code = '1';
}

int ClassOne::getSize()
{
return size;
}

char ClassOne::getCode()
{
return code;
}

ClassTwo::ClassTwo()
{
age = 2;
init = '2';
}

int ClassTwo::getAge()
{
return age;
}

char ClassTwo::getInit()
{
return init;
}
#endif
```

```cpp
// test_one.cpp CS210 sample F'03
// 11/7/03 - Ray S. Babcock

#include <iostream>
using namespace std;

#include "one.h"

int main(void)
{
ClassOne fred;
ClassTwo suzie;

int sum;

sum = fred.size + suzie.age;
cout << sum << endl;

return 0;
}
```

2. (14 points) Given the code to the right:

   (a) (4 points) what prints at the lines marked LINE ONE & LINE TWO?

   (b) (2 points) Given the code to the right what prints at the lines marked LINE THREE & LINE FOUR?

   (c) (2 points) Given the code to the right what prints at the lines marked LINE FIVE & LINE SIX?

   (d) (2 points) Given the code to the right what prints at the lines marked LINE SEVEN & LINE EIGHT?

   (e) (2 points) Given the code to the right what prints at the lines marked LINE NINE & LINE TEN?

   (f) (2 points) Given the code to the right what prints at the lines marked LINE ELEVEN & LINE TWELVE?

```
// two.h - CS210 sample F'03
// 11/7/03 - Ray S. Babcock

#ifndef TWO_H_
#define TWO_H_

class Two
{
  public:
    Two(int=3, int=5);
    ~Two();
    int SumGreen();
    int SumGrass();
    void PrintGreen(const char *);
    void PrintGrass(const char *);

  private:
    int * green;
    int * grass;
};

Two::Two(int green_value, int grass_value)
{
    green = new int[4];
    grass = new int[4];
    for(int i=0; i<4; i++)
    {
      green[i]=green_value;
      grass[i]=grass_value;
    }
}

Two::~Two()
{
delete [] green;
delete [] grass;
}
```

```
int Two::SumGreen()
{
  int sum(0);
  for(int i=0; i<4; i++)
   sum+=green[i];
  return sum;
}

int Two::SumGrass()
{
  int sum(0);
  for(int i=0; i<4; i++)
   sum+=grass[i];
  return sum;
}

void Two::PrintGreen(const char * label)
{
  cout << "green(" << label << "): ";
  for(int i=0; i<4; i++)
   cout << green[i] << " ";
  cout << endl;
}

void Two::PrintGrass(const char * label)
{
  cout << "grass(" << label << "): ";
  for(int i=0; i<4; i++)
   cout << grass[i] << " ";
  cout << endl;
}
#endif
```

```
// test_two.cpp - CS210 sample F'03
// 11/7/03 - Ray S. Babcock

#include <iostream>
using namespace std;

#include "two.h"

int main (void)
{
  Two first(1,2),
      second(3,4);

  first.PrintGreen("first");// LINE ONE
  first.PrintGrass("first");// LINE TWO

  first.PrintGreen("second");// LINE THREE
  first.PrintGrass("second");// LINE FOUR

  second.PrintGreen("exam2");// LINE FIVE
  second.PrintGrass("exam2");// LINE SIX

  Two third(6);
  Two fourth(third);

  third.PrintGreen("third");// LINE SEVEN
  third.PrintGrass("third");// LINE EIGHT

  fourth.PrintGreen("fourth");// LINE NINE
  fourth.PrintGrass("fourth");// LINE TEN

  fourth = second;

  fourth.PrintGreen("last");// LINE ELEVEN
  fourth.PrintGrass("last");// LINE TWELVE

  return 0;
}
```

3. (10 points) Given the code at the right: .

   (a) (6 points)What prints?

   (b) (4 points) Given the same vector with the same contents as shown in the code to the right. What would be in the interval [shine(0),shine(3))?

```
// five.cpp CS210 F'03
// 11/7/03 - Ray S. Babcock

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

int main(void)
{
  vector<int> shine;

  shine.push_back(5);
  shine.push_back(4);
  shine.push_back(3);
  shine.push_back(2);
  shine.push_back(1);

  vector<int>::const_iterator sun = shine.begin();
  ++sun;
  ++sun;

  cout << "A = " << *sun++ << endl;

  cout << "B = " << *sun << endl;

  return 0;
}
```