PIC 10A 1C Week 5b Problems. TA: Eric Kim.

1. The Declaration of Function-pendence

Based off of the following code snippets, infer what the function's declaration would be. The first is done for you.

Code	**Function Declaration**
`int a = 1; double b = 3.14;` `int result = myfn(a,b);`	int myfn(int arg1, double arg2);
`string name = "Nathan";` `char c = fn2(name, name);`	
`fn3(string("hi").substr(1,1), 42.0);`	

2. Classy Interfaces

Louis Reasoner is working on some classes. Given the class interface and usage, point out any possible errors. Assume that no libraries/namespaces have been included/set yet.

```
class Minion {
private:
  int myhealth;

  void perish();

  string sing();
public:
  int myid;

  Minion(int health);

  void attack(Minion victim);

  void take_damage(int
damage);

  int get_health();
};
```

```
int main() {
  Minion bob = Minion(10);

  Minion joe;

  attack(bob);

  bob.take_damage(2);

  cout << 'Bob health: '
    << bob.get_health() << endl;

  cout << "Bob attacks back! "
    << bob.attack(joe);

  joe.take_damage();

  cout << "Joe health: " << joe.myhealth;

  cout << "Bob ID: " << bob.myid;
  return 0;
}
```

3. Inferring the Interfaces

Given the following code snippet, infer what the class interface for the Goblin class is.

```
Goblin gunth("Gunth");
Hero arthur("Arthur");
int dmg = gunth.attack(arthur);
cout << gunth.get_name() << " attacked " << arthur.get_name() << "!" <<
endl;
cout << "Damage: " << dmg << endl;
cout << "The goblin sings the song of his people: " << gunth.sing();
gunth.restore_health();
cout << "The goblin restored health!\n");
return 0;
```

```
class Goblin {
// FILL ME IN!









}
```

4. Fizzbuzz Lite

Write a program that asks the user for an integer. If the integer is divisible by 3, display "Fizz". If the integer is divisible by 5, display "Buzz". If it's divisible by both 3 and 5, then print "Fizzbuzz". Otherwise, simply display the number.
Hint: To check if an integer `a` is divisible by an integer `b`, use the mod operator. "`(a % b) == 0`" is true when `a` is divisible by `b`.

YOUR CODE HERE

5. Don't Lose Your Head!

Recall that we use header guards (#ifndef, #define, #endif) to avoid accidentally defining variables/classes multiple times. Consider the following header files:

File: myheader1.h

```
#ifndef _MYHEADER1_
#define _MYHEADER1_
int MY_X = 100;
#endif
```

File: myheader2.h

```
int MY_X = -2;
```

```
#include <iostream>
#include "myheader2.h"
int main() {
    std::cout << "MY_X is: " << MY_X;    return 0;
}
```

What is the output of the program? If it crashes, describe the crash.

Next, consider the following header files:

File: h1.h
```
#include "h2.h"
#ifndef _H_
#define _H_
int MY_X = 100;
#endif
```

File: h2.h
```
#ifndef _H_
#define _H_
int MY_X = -50;
#endif
```

```
#include <iostream>
#include "h1.h"
int main() {
    std::cout << "MY_X is: " << MY_X;    return 0;
}
```

What is the output of the program? If it crashes, describe the crash.

6. Static Cling

For the following, insert **at most one** `static_cast` to make the expression evaluate to the desired value. Note that `static_cast` may not be necessary!

Code:	**Desired Output:**
`cout << (3.0 / 2) + 1;`	2
`cout << (6 / 3) + 1;`	3.0
`cout << (2/4) + 0.5 + (5/4);`	1.5

7. Storage Wars

Recall that, in a computer, there are three places to store data: primary storage (ie RAM), registers, and secondary storage (ie hard disk). Compare and contrast these three components. In particular, what are the strengths/weaknesses of each component?