[**Open book, handouts, and notes**] Show all of your work clearly in the space provided or on the additional page at the end of the exam. If the additional page is used, clearly identify to which exam question it is related. Be sure to **read each problem carefully**. You should answer all 3 questions. Due to time limitations, you are not required to comment your code on this exam. Note that the exam is double sided.

1. (10 points) Briefly, in your own words, describe how *pointers* and *iterators* differ. Use examples when appropriate.

2. (25 points) Write the header file associated with a Time class that will allow Time objects to be declared and used as shown in the code below. Note: You should indicate all the appropriate preprocessor directives (#include files, etc). Write **only** the function **prototypes** for the functions **used in the code below**. You **do not** need to **implement** the functions.

```
#include <iostream>
#include "Time.h"

int main()
{
  Time appointment;
  const Time bedtime(22); // Sets time to 10pm (22:00:00 military time)
  std::cout << "Please enter the time of your appointment ";
  std::cin >> appointment;
  std::cout << "If your appointment lasts thirty minutes, it will be done at "
            << appointment + 30 << std::endl;
  int freeTime = bedtime - 30 - appointment;
  std::cout << "That will give you " << freeTime << " minutes until bedtime."
  return 0;
}
```

3. Consider the following class definition for complex numbers:

```
#include <iostream>
using std::istream;
using std::ostream;

class Complex {
public:
  Complex();
  Complex(double rl, double im=0.0);
  Complex(const Complex& rhs);
  ~Complex();
  Complex& operator=(const Complex& rhs);
  void display(ostream& os);
  bool read(istream& is);
  double getReal();
  Complex operator+(Complex rhs);
  Complex operator-(const Complex rhs);
  Complex operator-(double rhs);
  Complex& operator*=(Complex rhs);
private:
  double real;
  double imaginary;
};
Complex operator+(double lhs, const Complex& rhs);
```

(a) (15 points) List the line numbers for all of the functions that should have been defined as `const` functions. (For partial credit, be sure to explain your reasoning.)

(b) (10 points) Identify all of the lines containing errors in the following code (based on the Complex class given previously). Note: there are no more than three errors, and there may not be any errors.

```
#include "Complex.h"
2
int main()
4 {
  Complex a;
6   Complex b(0,1);
  Complex c=2.0;
8   std::cout<<"Enter␣a␣complex␣number␣";
  std::cin >> a;
10  c = 3.2 + b;
  std::cout << c;
12  c += a;
  c = b - 7.25;
14  a = c + 0.52;

16  return 0;
}
```

(c) (10 points) Write the getReal member function.

(d) (15 points) Write the *= operator (line 18).

(e) (15 points) Implement the display member function so that it sends the complex number to the output stream. Here are some examples of what the results should look like:

$3.2 + i8.1$
$1.9 - i0.3$
17.2
$-i8.8$
$i6.1$
$-14.72 + i5$

Additional work area for any problem. Clearly identify to which problem the work on this page is related.