CSCI 135 Software Design and Analysis, C++
Lab 8
Solution

Saad Mneimneh
Hunter College of CUNY

Lab A: Skolem
An infinite Skolem sequence $a[0], a[1], a[2], \dots$ satisfies the following two conditions:

- for every $n \in N$, there exist exactly two integers i and j such that $a[i] = a[j] = n$. Furthermore, $i - j = n$.
- for every $n < m$, if i and j are the smallest such that $a[i] = n$ and $a[j] = m$, then $i < j$.

Here are the first few terms:

$$1\ 1\ 2\ 3\ 2\ 4\ 3\ 5\ 6\ 4\ 7\ 8\ 5\ 9\ 6\ \dots$$

Given an array of size k, fill the array with the first k terms of the infinite Skolem sequence. *Hint*: Initialize the array to zeros. Then for every n in increasing order, find the first spot that is available, say i, and assign $a[i]$ and $a[i+n]$ the value n. But make sure not to exceed the boundary of the array.

```
void SkolemFill(int * a, int k) {...}
```

Solution:

```
void SkolemFill(int * a, int k) {
  for (int i=0; i<k; i=i+1)
    a[i]=0; //spot is available
  int n=1;  //next number
  int i=0;  //next index
  while (i<k) {
    a[i]=n;
    if (i+n<k)
      a[i+n]=n;
    n=n+1;
    while (i<k && a[i]!=0) //look for an available spot
                           //using short-circuit evaluation
      i=i+1;               //if i>=k, a[i] will not be checked
  }
}
```

Lab B: Imaginary numbers
Consider the following class for imaginary numbers:

```
class Im {
  double r;
  double i;

 public:
  Im() {...}
  Im(double rl) {...}
  Im(double rl, double imgnr) {...}

  void set(double rl, double imgnr) {...}

  double real() {//returns the real part}
  double im() {//returns the imaginary part}
  bool isIm() {//returns true iff imaginary part is not zero}

  void print() {cout<<r<<"+i"<<i;}

  Im add(Im n) {...}
  Im sub(Im n) {...}
  Im mul(Im n) {...}
  Im div(Im n) {...}
};
```

Complete the implementation of the class.
Note:

$$(a+ib)+(c+id)=(a+c)+i(b+d)$$
$$(a+ib)-(c+id)=(a-c)+i(b-d)$$
$$(a+ib)(c+id)=(ac-bd)+i(ad+bc)$$
$$(a+ib)/(c+id)=(a/l+ib/l)(c-id)$$

where $l=c^2+d^2$.

Solution:

```
//helper
double square(double x) {
  return x*x;
}

class Im {
  double r;
  double i;

 public:
  Im() {set(0,0);}
  Im(double rl) {set(rl,0);}
  Im(double rl, double imgnr) {set(rl, imgnr);}

  void set(double rl, double imgnr) {r=rl; i=imgnr;}
```

```
  double real() {return r;}
  double im() {return i;}
  bool isIm() {return (i!=0);}

  void print() {cout<<r<<"+i"<<i;}

  Im add(Im n) {return Im(r+n.r, i+n.i);}
  Im sub(Im n) {return Im(r-n.r, i-n.i);}
  Im mul(Im n) {return Im(r*n.r-i*n.i, r*n.i+i*n.r);}
  Im div(Im n) {
    l=square(n.r)+square(n.i);
    return mul(Im(r/l, i/l), Im(n.r, -n.i));
  }
};
```