Tutorial Problem Set #5

Due: Wednesday, October 23, 2024, 11:59 PM

Policy

- Piazza questions on tutorial problems will be ignored or deleted. Questions will only be answered in your assigned tutorial section.
- Sample executables can be found in your 1249 git repository directory (run `git pull`).
- Completing the problem set will reduce the weight of the final exam by 0.5%. To complete a problem set, you must pass at least 50% of the secret tests.
- You may assume all input is valid. Tutorial problems **NEVER** require checking for invalid inputs.

Question 1

You've been playing with common mathematical sequences over integers for fun, and decide you'd like to be able to iterate over the results of such functions. Naturally, since you're taking CS246, you decide to write an iterator class in C++ with a constructor that takes in an `int` that is the location in the sequence to begin, and a pointer to a single-parameter function that takes in an `int` and returns an `int`.

Consider the following partial program:

```
int fib( int n ) {
    return n <= 2 ? 1 : fib( n-1 ) + fib( n-2 );
}

int pow2( int n ) {
    return n == 0 ? 1 : 2 * pow2( n-1 );
}

void printRange( SeqIt & begin, SeqIt & end ) {
    while ( begin != end ) {
        cout << *begin << " ";
        ++begin;
    }
    cout << endl;
}

int main() {
    // Note: this is how to declare a function pointer named foo to
    // a function that returns an int and has one int parameter.
    // It's not necessary for the function call, but is included
    // to remind you of how to write a function pointer :)
    int (*foo)(int);
    foo = fib;
    SeqIt myStart{ 5, foo };
    SeqIt myEnd{ 10, foo };
    printRange( myStart, myEnd );
    myStart = SeqIt{ 0, pow2 };
    myEnd = SeqIt{ 10, pow2 };
    printRange( myStart, myEnd );

    // Iterator comparisons:
    cout << boolalpha << ( SeqIt{5, fib} != SeqIt{5, fib} ) << endl;
    cout << boolalpha << ( SeqIt{5, fib} != SeqIt{5, pow2} ) << endl;
    cout << boolalpha << ( SeqIt{10, fib} != SeqIt{5, fib} ) << endl;
} // main
```

You must write the iterator class `SeqIt` (Sequence Iterator) such that the above program compiles and prints out:

```
5 8 13 21 34
1 2 4 8 16 32 64 128 256 512
false
true
true
```

Submission

Submit your solution files `seqit.cc` and `seqit-impl.cc` in a ZIP file called `tut05.zip` to Marmoset.