Matrix* - a library for the book from the authors

Drill.cpp – the drill exercise from Chapter 24

1. Print the size of a **char**, a **short**, an **int**, a **long**, a **float**, a **double**, an **int***, and a **double*** (use sizeof, not <limits>).
2. Print out the size as reported by **sizeof** of **Matrix<int> a(10)**, **Matrix<int> b(100)**, **Matrix<double> c(10)**, **Matrix<int,2> d(10,10)**, **Matrix<int,3> e(10,10,10)**.
3. Print out the number of elements of each of the **Matrixes** from 2.
4. Write a program that takes **ints** from **cin** and outputs the **sqrt()** of each **int**, or "no square root" if **sqrt(x)** is illegal for some **x** (i.e., check your **sqrt()** return values).
5. Read ten floating-point values from input and put them into a **Matrix<double>**. **Matrix** has no **push_back** so be careful to handle an attempt to enter a wrong number of doubles. Print out the **Matrix**.
6. Compute a multiplication table for [0,n)*[0,m) and represent it as a 2D **Matrix**. Take **n** and **m** from **cin** and print out the table nicely (assume that **m** is small enough that the results fit on a line).
7. Read ten **complex<double>**s from **cin** (yes, **cin** supports >> for **complex**) and put them into a **Matrix**. Calculate and output the sum of the ten complex numbers.
8. Read six **ints** into a **Matrix<int,2> m(2,3)** and print them out.