Physics 234: Exercise 3

1. (a) Of the $N^2 - N = N(N-1)$ elements not on the main diagonal, half can be thrown away. Hence, at least $N^2 - N(N-1)/2 = N(N+1)/2$ must be kept.

   (b) The indexing rules $I(i,j)$ for the symmetric matrix case are as follows.

	row	column
upper	$j + i(2N-1-i)/2$	$i + j(j+1)/2$
lower	$j + i(i+1)/2$	$i + j(2N-1-j)/2$

2. (a) The argument from the symmetric case still holds, except that now all the elements on the main diagonal can also be tossed out. Hence, only $N(N+1)/2 - N = N(N-1)/2$ elements are needed.

   (b) Here are the indexing rules for an antisymmetric matrix.

	row	column
upper	$j - i - 1 + i(2N-1-i)/2$	$i + j(j-1)/2$
lower	$j + i(i-1)/2$	$i - j - 1 + j(2N-1-j)/2$

3. It's helpful to express the value of the exponent in base two: $19 = 00010011_2$. From the sequence of successive squares, $s_0 = a$, $s_1 = s_0^2$, $s_2 = s_1^2$, $s_3 = s_2^2$, $s_4 = s_3^2$, ..., it's straightforward to construct $a^{21} = s_4 s_1 s_0$. The direct translation of this process into C++ code requires six multiplications:

```
const double a = ....;
const double a2 = a*a;
const double a4 = a2*a2;
const double a8 = a4*a4;
const double a16 = a8*a8;
const double a19 = a16*a2*a1;
```

   In general, to compute a^p requires roughly $\log_2 p$ multiplications. This is a big improvement over the $p-1$ that are needed to write `a*a*a*···*a` in the most naive way.

4. The usual approach is to construct a nested monomial factorization of the polynomial.

$$\begin{aligned} a^4 - 6a^3 + 11a^2 - 4a + 1 &= (a^3 - 6a^2 + 11a - 4)a + 1 \\ &= ((a^2 - 6a + 11)a - 4)a + 1 \\ &= (((a - 6)a + 11)a - 4)a + 1 \end{aligned}$$

The corresponding computation requires seven algebraic operations:

```
const double result = (((a-6)*a+11)*a-4)*a+1;
```

But this isn't quite the most efficient way. Consider the following rewriting.

$$\begin{aligned} a^4 - 6a^3 + 11a^2 - 4a + 1 &= (a^4 - 6a^3 + 11a^2 - 6a + 1) + 2a \\ &= (a^2 - 3a + 1)^2 + 2a \\ &= \big[a(a - 3) + 1\big]^2 + 2a \end{aligned}$$

Clearly, the computation can be carried out with only six operations (although at the cost of a `double` temporary):

```
const double tmp = a*(a-3)+1;
const double result = tmp*tmp + 2*a;
```