ES112 Summer 2003-2004
First Midterm Examination

1 Calculating Square Roots *(40 points)*

Consider the series defined as:

$$a_{n+1} = \frac{a_n^2 + x}{2a_n}, \quad a_0 = 1,\ x \geq 0$$

The interesting thing is the following:

$$\lim_{n \to \infty} a_n = \sqrt{x}$$

In other words, successive terms of the series converge to the square root of x. And, as you can see, each successive term can be computed just by elementary arithmetic operations.

Your task is to write a C function that will take a double x as an argument, and return its square root as a double. You are not allowed to use any math library functions. (The idea is to write one!) Using the formula above is a good idea. Of course, it is not possible to calculate an infinite number of terms. However, your result should be precise such that:

Assuming your function returns s as the square root of x; the precision should be so that:

$$\left|s^2 - x\right| < 10^{-15}$$

Last, but not least, the prototype of the function must be:

```
double square_root(double x);
```

2 Print an Integer in Reverse *(40 points)*

Write a C function that takes an unsigned integer n as an argument, and prints it on the screen *backwards*. You may *not* assume anything about the number of digits of n. The prototype of the function is:

```
void print_backwards(unsigned int n);
```

So, for instance, if the function is called with 22990, it should print 09922 on the screen, and if it is called with 123, it should print 321 on the screen. Also, if called with 0, it should print 0 on the screen.

3 Divisibility by Four *(20 points)*

Write a C function that takes an integer n as an argument, and returns an integer which is 1 if n is divisible by four, and 0 if n is not divisible by four. The prototype of the function is:

```
int divisible_by_four(int n);
```

You are not allowed to use addition, subtraction, multiplication, division or the modulus operator. Loops are not allowed either. However, you are free to use bitwise operators.