MATH 416
MID-TERM EXAM
SOLUTIONS

1. (a) *Write down a precise definition for the meaning of $f(n) = \omega(g(n))$.*
Solution: For every constant $C > 0$ there exists a constant n_0 such that $0 \leq Cg(n) < f(n)$ for all $n \geq n_0$.
(b) *Let $f(n) = (\sqrt{8})^{\lg n}$ and $g(n) = n(\lg n)^3$. Determine (with a brief explanation) whether $f(n) = o(g(n))$, $f(n) = \Theta(g(n))$, $f(n) = \omega(g(n))$ or none of the three.*
Solution: We have

$$f(n) = (\sqrt{8})^{\lg n} = (2^{3/2})^{\lg n} = (2^{\lg n})^{3/2} = n^{3/2}$$

So

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = \lim_{n\to\infty} \frac{n^{3/2}}{n(\lg n)^3} = \lim_{n\to\infty} \frac{n^{1/2}}{(\lg n)^3} = \infty.$$

(In the book it is noted that $\lim_{n\to\infty} \frac{n^a}{(\lg n)^b} = \infty$ if a, b are positive constants.) Since $g(n)$ is also clearly assymptotically nonnegative, we have that $f(n) = \omega(g(n))$.

2.
(a) (10 points) Use the **Master Theorem** to find the solution to the following recurrence relations

$$T(n) = 3T(n/2) + n^2 \lg n.$$

The answer should be of the form $T(n) = \Theta(\qquad)$. Show how you apply the Master Theorem.
Solution:$a = 3, b = 2$ and$f(n) = n^2 \lg n$. $\log_2 3 < \log_2 4 = 2$. Choose ϵ such that $0 < \epsilon \leq 2 - \log_2 3$. Then we have $f(n) = \Omega(n^{\log_2 3+\epsilon})$. We are in the third case but we still have to check the regularity condition: there should exists constants $c < 1$ and $n_0 > 0$ such that $3f(n/2) \leq cf(n)$ for all $n \geq n_0$. We have

$$3f(n/2) = \frac{3n^2}{4} \lg n/2 < \frac{3}{4} n^2 \lg(n) = \frac{3}{4} f(n).$$

This shows that we can take $c = \frac{3}{4}$ and $n_0 = 1$. We can apply the Master theorem which says that $T(n) = \Theta(f(n)) = \Theta(n^2 \lg n)$.
(b) (5 points) What is the running time (in terms of arithmetic operations) of multiplying two $n \times n$ matrices with Strassen's method?
Solution: $\Theta(n^{\lg 7})$.

Date: fall 2002.

(c) (5 points) What is the running time (in terms of arithmetic operations) of finding the LUP decomposition of an $n \times n$ matrix (with the method described in the book).
Solution: $\Theta(n^3)$.

3. Below is the pseudocode for `Bubblesort`. Bubblesort takes an array $A[1 \dots n]$ and sorts it in place.

```
Bubblesort(A)
for i ← 1 to length[A]
do for j ← length[A] downto i + 1
   do if A[j] < A[j − 1]
      then exchange A[j] ↔ A[j − 1]
```

(a) (10 points) What is the worst case running time of `Bubblesort` when A is an array of size n?
Solution: The `if` statement is executed for all i, j with $1 \leq i \leq n$ and $i+1 \leq j \leq n$. It is therefore executed $(n-1)+(n-2)+\cdots+1+0 = n(n-1)/2$ times. Every other line in the program is clearly executed at most $O(n^2)$ times. This shows that the running time is $\Theta(n^2)$. (worst case, best case and average case!)

(b) (10 points) Write down a loop invariant for the outer loop. Your loop invariant should be correct and useful for the correctness proof of the algorithm. However, You do not have to prove that your loop invariant is a loop invariant. Nor do you have to prove the correctness of the algorithm.
Solution: At the beginning of each iteration of the outer loop we have that $A[j] \leq A[k]$ for all j, k with $0 \leq j \leq i-1$ and $j \leq k$. Moreover, the entries $A[1 \dots n]$ are equal to the original $A[1 \dots n]$ but possibly in a different order.

4. (20 points) Use the *Extended Euclid's algorithm* to find integers x and y such

$$23x + 16y = 1.$$

Show your work.
Solution: We use the extended GCD algorithm:

a	b	d	x	y
23	16	1	7	-10
16	7	2	-3	7
7	2	3	1	-3
2	1	2	0	1
1	0			

We can take $x = 7, y = 10$ to get $23 \cdot 7 - 16 \cdot 10 = 1$.

5. (20 points) Suppose that the Discrete Fourier Transform of some polynomial

$$f(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3$$

is

$$(2, 3 - 3i, 4, 3 + 3i).$$

In other words, $f(1) = 2$, $f(i) = 3 - 3i$, $f(-1) = 4$ and $f(-i) = 3 + 3i$. Compute the coefficients a_0, a_1, a_2, a_3 of $f(x)$ (using the inverse discrete Fourier transform). Show your work.

Solution: We have to compute $\mathrm{DFT}^{-1}(2, 3 - 3i, 4, 3 + 3i)$. This is the same as $\frac{1}{4}\mathrm{DFT}(2, 3 + 3i, 4, 3 - 3i)$. We compute

$$\mathrm{DFT}(2, 4) = (2 + 4, 2 - 4) = (6, -2)$$

and

$$\mathrm{DFT}(3 + 3i, 3 - 3i) = ((3 + 3i) + (3 - 3i), (3 + 3i) - (3 - 3i)) = (6, 6i).$$

Then

$$\mathrm{DFT}(2, 3 + 3i, 4, 3 - 3i) = (6 + 6, -2 + (6i)i, 6 - 6, -2 + (6i)(-i)) = (12, -8, 0, 4).$$

so we get

$$\mathrm{DFT}^{-1}(2, 3 - 3i, 4, 3 + 3i) = \frac{1}{4}\mathrm{DFT}(2, 3 + 3i, 4, 3 - 3i) = \frac{1}{4}(12, -8, 0, 4) = (3, -2, 0, 1).$$

So we have $a_0 = 3, a_1 = -2, a_2 - 0, a_3 = 1$ and $f(x) = x^3 - 2x + 3$. We can check that the values $f(1), f(-1), f(i), f(-i)$ are correct.