MAD 3105 — Discrete Math 2

Section 1, Spring 1996. MWF 1:25–2:15. 222 MCH

Instructor: Bellenot.

The good doctor's Office: 002-B Love, Office Hours: MWF 2:30 - 3:15 or by appointment. Email addressed bellenot@cs.fsu.edu, bellenot@math.fsu.edu, or even bellenot@fsu.edu will get to the good doctor, but the short address 'bellenot' works on math or cs machines.

Eligibility: A grade of C- or better in Discrete Math 1 (MAD 3104).

Text: Dossey, Otto, Spence and Eynden, *Discrete Mathematics* 2^{nd} Edition.

Coverage: Chapters 5-8 and additional material as time allows.

Final: At 7:30 – 9:30 am Thursday, Apr 25, 1995. (Sort of late in the week.)

Tests: (3) Tentatively at Jan 31, Feb 28?(or Mar 6?) and Apr 10. No Makeup tests.

Quizes: Every Wednesday (except test days). No Makeup quizes.

Grades: 90% A, 80%B, 70%C, 60%D.

Relative Weights F = 2T and T = Q (F is 1/3, each T is 1/6 and Q is 1/6).

Homework and Attendance are required. Indeed attendance will be taken by checking off homework. It is the student's responsibility to see that homework is delivered on time. (The homework needs to be turned in even when the student is absent.) Likewise, being absence is not a valid reason for not knowing the next assignment.

Four or more late or missing homeworks is an automatic FAIL.

Fair Warning: *The good doctor teaches this course differently than other instructors. The order of material will be different and so will be the emphasis.*

Proofs: At the request of the computer science faculty this class contains proofs. Part of the grade of a proof will be based on the proof having the correct form (in addition to its being a proof or not).

The Web page for the class is "http://www.math.fsu.edu/~bellenot/class/dm2".

Rough Idea for Alternate 4.2.1

"Big Oh" Notation

This is an intuitive approach using limits. The use of limits here can also be done intuitively. Previous knowledge of limits is not assumed.

Consider the table below, $f(n)$ is the function,

n	$n^{\frac{1}{3}}$	$n^{\frac{1}{2}}$	$n \log n$	n^2	n^3	n^8	2^n	4^n	$n!$
1	1	1	0	1	1	1	2	4	1
2	1.3	1.4	1.4	4	8	256	4	16	2
3	1.4	1.7	3.3	9	27	6561	8	64	6
10	2.2	3.2	23	100	10^3	10^8	1024	10^6	3.6×10^6
20	2.7	4.5	60	400	8×10^3	2.6×10^{10}	10^6	10^{12}	2.4×10^{18}
100	4.6	10	460	10^4	10^6	10^{16}	1.3×10^{30}	1.6×10^{60}	9.3×10^{157}
1000	10	31.6	6900	10^6	10^9	10^{24}	10^{301}	1.2×10^{602}	4×10^{2567}

Most entries are approximate

The functions in the table are in increasing order (in terms of big oh) going from left to right. (Although the function n fits between $n^{\frac{1}{2}}$ and $n \log n$.) That is $\mathcal{O}(n^{\frac{1}{3}}) < \mathcal{O}(n^{\frac{1}{2}}) < \mathcal{O}(n) < \mathcal{O}(n \log n) < \mathcal{O}(n^2) < \mathcal{O}(n^3) < \mathcal{O}(n^8) < \mathcal{O}(2^n) < \mathcal{O}(4^n) < \mathcal{O}(n!)$. Note that this does not say $n^8 < 2^n$ for all values of n. (Certainly it isn't true for $n = 2$.) Big oh "measures" what happens for large values of n. (already by $n = 100$, it takes twice as many digits to write out 2^n as it does to write out n^8.)

Also big oh is a "rough measure", that is, $\mathcal{O}(n^8) = \mathcal{O}(13n^8)$. Multiplying a function by a positive constant does not change its big oh. After all, multiplying the entries of the n^8 column by 13 isn't going to help it catch up with 2^n.

There are two useful rules in the table. The first is the approximation $2^{10} = 1024 \sim 10^3$. Thus $2^{100} = (2^{10})^{10} = 10^{30}$ and $2^{24} = (2^{10})^2 \cdot 2^4 \sim 16 \times 10^6$. The second rule is hidden better. It is $\mathcal{O}(n \log n) = \mathcal{O}(\log(n!))$. (The log used in the table is the natural log.) More on this second rule later.

Well, it's time to give a way of determining when $\mathcal{O}(f) = \mathcal{O}(g)$ or $\mathcal{O}(f) < \mathcal{O}(g)$. The theorem below doesn't always do this for general functions $f(n)$ and $g(n)$. But it will work for the functions found in this book.

Theorem. *Suppose*

$$\lim_{n\to\infty} f(n) = \lim_{n\to\infty} g(n) = \infty \text{ and } \lim_{n\to\infty} \frac{f(n)}{g(n)} = L,$$

then if

$$\begin{array}{rll} \mathcal{O} < L < \infty & \text{we have} & \mathcal{O}(f) = \mathcal{O}(g) \\ \text{or if } L = 0 & \text{we have} & \mathcal{O}(f) < \mathcal{O}(g) \\ \text{or if } L = \infty & \text{we have} & \mathcal{O}(f) > \mathcal{O}(g). \end{array}$$

To say $\lim_{n\to\infty} f(n) = \infty$, just means intuitively if n is "infinitely large" then $f(n)$ is "infinitely large" or that $f(n)$ grows without bound as n gets big. To say $\lim_{n\to\infty} \frac{f(n)}{g(n)} = L$ means $\frac{f(n)}{g(n)}$ is close to L as n gets big. Let's do some examples to get the idea.

Examples

1. $\mathcal{O}(n) = \mathcal{O}(2n)$

   since $\lim_{n\to\infty} n = \lim_{n\to\infty} 2n = \infty$ and $\lim_{n\to\infty} \frac{n}{2n} = \lim_{n\to\infty} \frac{1}{2} = \frac{1}{2}$.

2. $\mathcal{O}(n^2) = \mathcal{O}(7n^2 + 100n + 13)$

   $$\frac{7n^2 + 100n + 13}{n^2} = 7 + \frac{100}{n} + \frac{13}{n^2} \to 7$$

   note that as $n \to \infty$, both $\frac{100}{n}$ and $\frac{13}{n^2}$ get small.

3. $\mathcal{O}(n^8 + 5n^3) = \mathcal{O}\left(\frac{1}{2}n^8 + 6\right)$

   $$\frac{n^8 + 5n^3}{\frac{1}{2}n^8 + 6}\left(\frac{\frac{1}{n^8}}{\frac{1}{n^8}}\right) = \frac{1 + \frac{5}{n^5}}{\frac{1}{2} + \frac{6}{n^8}} \to \frac{1}{\frac{1}{2}} = 2$$

   "The trick" is divide both top and bottom by the highest power of n.

4. $\mathcal{O}(n) < \mathcal{O}(n \log n) < \mathcal{O}(n^2)$

   $$\frac{n}{n \log n} = \frac{1}{\log n} \to 0$$

   $$\frac{n \log n}{n^2} = \frac{\log n}{n} \to 0$$

The fact that $\frac{\log n}{n} \to 0$ is usually proved in calculus classes. Intuitively speaking, we see that since $n = 10^{\log n}$, it takes $\log n$ digits to write n and so n is much bigger than $\log n$.

In fact, $\mathcal{O}(\log n) < \mathcal{O}(n^k)$, for any $k > 0$. Thus $\mathcal{O}(n \log n) < \mathcal{O}(n^{1+k})$, for any $k > 0$ and in particular when $k = 1$.

5. $\mathcal{O}(2^n) < \mathcal{O}(4^n)$

$$\lim_{n\to\infty} \frac{2^n}{4^n} = \lim_{n\to\infty} \left(\frac{2}{4}\right)^n = 0 \text{ since } \frac{2}{4} = \frac{1}{2} < 1.$$

6. If $A > 1$, then $\mathcal{O}(n) < \mathcal{O}(A^n)$.

Taking the limit of $\frac{n}{A^n}$ is easy if one knows enough calculus. However we can still do it with more work.

Lemma 1. *If $B > 1$ and N is large enough so that $B > 1 + \frac{1}{N}$ and let $K = \frac{N}{8^N}$, then for $n \geq N$, $n \leq KB^n$.*

Proof. By induction. When $n = N$, $K = \frac{N}{B^N}$ or $N = KB^N$. Assume $n \leq KB^n$ is true. Multiply by B getting $Bn \leq KB^{n+1}$. Now $B > 1 + \frac{1}{N}$ so $Bn > n + \frac{n}{N}$. But $\frac{n}{N} \geq 1$, so $Bn \geq n + 1$ and $n + 1 \leq KB^{n+1}$.□

Now $\dfrac{n}{A^n} = \dfrac{n}{B^n}\dfrac{B^n}{A^n}$ for $1 < B < A$. so $\dfrac{n}{A^n} \leq K\left(\dfrac{B}{A}\right)^n$ for $n \geq N$ and thus $\dfrac{n}{A^n} \to 0$.

7. If $A > 1$, then $\mathcal{O}(n^3) < \mathcal{O}(A^n)$ let $B = A^{\frac{1}{3}} > 1$ thus $\dfrac{n}{B^n} \to 0$ by 6 so that $\left(\dfrac{n}{B^n}\right)^3 = \dfrac{n^3}{B^{3n}} = \dfrac{n^3}{A^n} \to 0$.

8. $\mathcal{O}(10^n) < \mathcal{O}(n!)$

Let $N = 20$ and note if $n > N$ $\dfrac{10^n}{n!} \leq \dfrac{10^N}{N!}\left(\dfrac{1}{2}\right)^{n-N} \to 0$. (Prove it by induction.)

Problems

1. Show $\mathcal{O}(n+1) = \mathcal{O}(10n+7) = \mathcal{O}(n+\log n) = \mathcal{O}(n)$

2. Show $\mathcal{O}(n^3+n^2+n+1) = \mathcal{O}(n^3-13) = \mathcal{O}(n^3)$

3. Show $\mathcal{O}(n^{\frac{1}{2}}) < \mathcal{O}(n)$

4. Show $\mathcal{O}(\sqrt{n^2+1}) = \mathcal{O}(n)$

5. Show $\mathcal{O}(n^{100}) < \mathcal{O}(n^{101}+\sqrt{n})$

6. Show $\mathcal{O}(2^n) < \mathcal{O}(3^n)$

7. Show $\mathcal{O}(n2^n) < \mathcal{O}(2^n)$

8. Show $\mathcal{O}(n2^n) < \mathcal{O}(3^n)$

9. Show $\mathcal{O}(n^{100}) < \mathcal{O}(2^n)$

10. Show $\mathcal{O}(100^n) < \mathcal{O}(n!)$

11. How are the big oh's of the following related? $\sqrt{n}$, $n\log n$, $n\sqrt{n}$, 2^n, $\log n$, $n2^n$, $2^n \log n$, n^2, $n^{\frac{1}{5}}$

12. If there are two programs P1 and P2 that do the same thing and P1 runs in $\mathcal{O}(f)$ and P2 runs in $\mathcal{O}(g)$ and $\mathcal{O}(f) < \mathcal{O}(g)$, does this mean P1 will always be faster than P2? Why or why not?

13. Approximate 2^{100}, 2^{18}, 2^{16}, 2^{32} using $2^{10} \sim 10^3$.

14. The following formula is in advanced calculus textbooks

$$1 \leq \frac{n!}{\sqrt{2n\pi}(\frac{n}{e})^n} \leq 1 + \frac{1}{12n-1}.$$

Use it to show $\mathcal{O}\left(\left(\frac{n}{e}\right)^n\right) < \mathcal{O}(n!) < \mathcal{O}(n^n)$ and that $\mathcal{O}(n\log n) = \mathcal{O}(\log n!)$. (Note $\mathcal{O} * \log_b n) = \mathcal{O}(\log_a n)$ if $a, b > 1$.)

Solve the divide-and-conquer relations using a change of variables.

(a) $a_n = 5a_{n/2} + 4$ where $a_1 = 0$ and $n = 2^k$ for $k \geq 0$.

(b) $a_n - 2a_{n/3} = 4$ where $a_1 = 5$ and $n = 3^k$ for $k \geq 0$.

(c) $a_n - 3a_{n/8} = 2n$ where $a_1 = 1$ and $n = 8^k$ for $k \geq 0$.

(d) $a_n - 5a_{n/3} = n$ where $a_1 = 5/2$ and $n = 3^k$ for $k \geq 0$.

(e) $a_n - 5a_{n/5} = n$ where $a_1 = 7$ and $n = 5^k$ for $k \geq 0$.

Strong Induction Problems.

```
Boolean BinSearch ( int key, SortedListofIntegers s )
if the length of s is zero
  return false
else if the length of s is one
  return true if the list is key otherwise return false
else let k be the middle element of s, s1 the list before k and s2 the list after.
  if key is k
    return true
  else if key < k
    return BinSearch ( key, s1 )
  else
    return BinSearch ( key, s2 )
```

1. Prove by stong induction on the length of the list s that BinSearch halts.
2. Assuming BinSearch halts, prove by stong induction on the length of the list s that BinSearch correctly determines if key is in the list.

```
List MergeSort ( List s )
if the length of the list is less than or equal 1
  return s
else divide the list into halves s1 and and s2 (or as near halves as possible)
  return Merge( MergeSort(s1), MergeSort(s2) )
```

3. Assuming Merge halts, prove by stong induction on the length of the list s that MergeSort halts.
4. Assuming MergeSort halts and Merge is correct, prove by stong induction on the length of the list s that MergeSort returns a sorted list.

```
VeryLongInteger Multiply ( VeryLongInter x, VeryLongInteger y)
Let n be the number of digits in the longest of x and y, and d = n/2.
If n less than or equal 1
  return x * y
else find x1, x2, y1, y2 each have no more than d digits so that
  x = x1 · 10^d + x2 and y = y1 · 10^d + y2
  let m1 = Multiply ( x1, y1 )
  let m2 = Multiply ( x2, y2 )
  let m3 = Multiply ( x1 + x2, y1 + y2 ) - m1 - m2
  return m1 · 10^(2d) + m3 · 10^d + m2
```

5. Prove by stong induction on the number of digits n that Multiply halts.
6. Assuming Multiply halts, prove by stong induction on the number of digits n that Multiply correctly multiplies the two numbers x and y.

MAD 3105 Discrete Math 2 **Test 1** 31 Jan 1996

Show **ALL** work for credit; be neat; and use only **ONE** side of each page of paper.

1. Tell how many systems of distinct representatives the given sequence of sets has

A. $\{1,2,4\}\{2,4\},\{3\},\{2,3\}$
B. $\{1,4\},\{2\},\{2,3,5\},\{1,2,4\},\{1,2\}$
C. $\{1,2,3,\ldots n\},\{1,2,3,\ldots n\},\{1,2,3,\ldots n\}$
D. $\{1,2,3,\ldots n\},\{n+1,n+2,\ldots n+m\},\{n+m+1,n+m+2,\ldots,n+m+k\}$

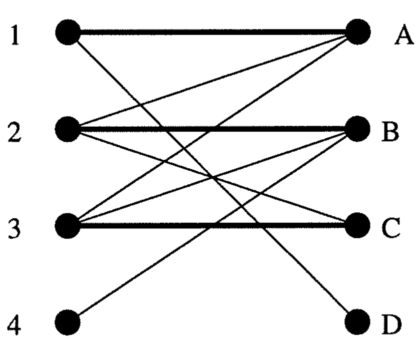
2. Binominal coefficients

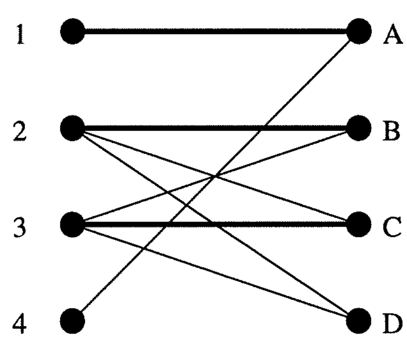
A. Draw Pascal's triangle until you get to the row needed for $(x+y)^7$
B. Expand $\binom{3n}{3}$ as a polynominal and simplify.

3. Here is a state table with ouput. Draw the transition diagram and list the output for the input sequence 0111100101 assuming A is the initial state.

Input	A	B	C	D	A	B	C	D
0	A	A	B	C	s	d	d	d
1	B	C	D	D	d	d	d	s

4. and 5. There are two bi-partite graphs below each with a matching (independent) set indicated by the bold edges. For each graph, change the graph into a matrix of zero's and one's with the correct one's starred. Procede with the algorithm of section 5.3 carefulling labeling the matrix, until either the end of step 4 or step 6 which ever comes first. If the algorithm stops in step 4, write down the vertices in the minimal cover obtained in step 4. If the algorithm ends in step 6, re-draw the bi-partite graph indicating the new matching.





6. Solve the bottleneck problem below. Show find the minimum completion time and show any smaller time is not a solution.

$$\begin{bmatrix} 3 & 5 & 5 & 3 & 8 \\ 4 & 6 & 4 & 2 & 6 \\ 4 & 6 & 1 & 3 & 6 \\ 3 & 4 & 4 & 6 & 5 \\ 5 & 7 & 3 & 5 & 9 \end{bmatrix}$$

7. Given $s_1 = 5$ and $s_n = 3s_{n-1} - 2^{n-1}$ for $n \geq 2$. Prove by induction $s_n = 3^n + 2^n$ for $n \geq 1$.

8. Devise a finite state machine (show the transition diagram) with inputs $I = \{0,1\}$ which accepts every string but those that contain 4 consecutive inputs (bits) of the form '0101'.

MAD 3105 Discrete Math 2 **Test 2** 6 Mar 1996

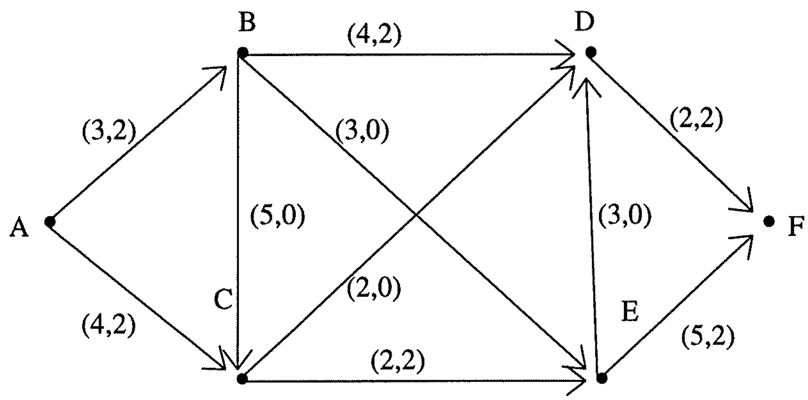
Show **ALL** work for credit; be neat; and use only **ONE** side of each page of paper.

1. A lumber company owns 7000 birch trees. Each year the company plans to harvest 12 percent of its trees and then plant 600 new ones. Write a recurrence relation and initial conditions for the number s_n of trees at the end of year n.

2. Given $s_n - s_{n-1} - 6s_{n-2} = 0$. Write the general solution to this recurrence relation. Explicitly find the solution which also satisfies the initial conditions $s_0 = 2, s_1 = 2$.

3. Given $s_n + 3s_{n-1} = 5$. Write the general solution to this recurrence relation. Explicitly find the solution which also satisfies the initial conditions $s_0 = 4$.

4. For the network below, use the flow augmentation algorithm to find a maximal flow, and its value. Use the text's convention for labeling vertices in alphabetical order when there is a choice.



5. Find the smallest sum of an independent set of entries from the matrix below and indicate an independent set of entries that has this smallest sum.

$$\begin{bmatrix} 8 & 2 & 4 & 6 \\ 3 & 1 & 7 & 5 \\ 4 & 6 & 5 & 3 \\ 4 & 5 & 2 & 1 \end{bmatrix}$$

6. Write a recurrence relation and initial conditions for the number s_n of n-bit strings having no four consecutive zeros.

7. Prove by strong induction that the algorithm NPF halts. [NPF computes the number of primes (counting repetitions) in the prime factorization of n.]

```
integer NPF ( integer n )
if n less than or equal to 1
  return 0
else if n is prime
  return 1
else let p be the smallest integer greater than 1 s.t.  p divides n and let q be n/p
  return NPF ( p ) + NPF ( q )
```

8. It can be shown that for an input of length n, the run time, s_n, of some divide and conquer algorithm satisfies the recurrence relation $s_n = 2s_{n/2} + 3n$ and $s_1 = 1$. Solve the recurrence by using the substitution $n = 2^k$.

MAD 3105 Discrete Math 2 **Test 3** 10 Apr 1996

Show **ALL** work for credit; be neat; and use only **ONE** side of each page of paper. Start problems on **LEFT** side of the paper only.

1. Big O.
 A. Prove $O(4n^2 - 5 + 13n^3) = O(100n^3 - n + n^2 - 3)$.
B&C. Arrange in increasing order:

$$O(\log n), O(n!), O(n^6), O(\sqrt{n}), O(3^n), O(n), O(n\sqrt{n}), O(n^n), O(n \log n), O(2^n).$$

2. Determine the successor in lexicographic order.
 A. For the permutations $(5,3,6,4,2,1)$ and $(2,3,1,6,5,4)$.
 B. For the five-element subsets of $\{n : 1 \leq n \leq 9\}$ after each of $\{1,3,5,6,7\}$, $\{1,2,3,8,9\}$ and $\{3,5,7,8,9\}$.

3. & 4. How many of 5 card poker hands are there?
 A. With 3 spades and 2 clubs?
 B. With 2 Jacks, a five, the three of hearts and a seven?
 C. With a full house (3 of a kind and an another pair)?
 D. At least one pair?
 E. With exactly one Queen and exactly 4 hearts?

5. & 6. How many ways are there of putting 22 balls into 7 boxes,
 A. If the balls are distinct?
 B. If the balls are identical?
 C. If the balls are identical and every box has at least one ball?
 D. If the balls are identical and no box has more than 19 balls
 E. Prove some box will have at least 4 balls after the balls have been put into the boxes.

7. Give network counterexamples to each statement below.
 A. If $|F| = 0$, then every edge has zero flow.
 B. If $F(\mathcal{T}, \mathcal{S}) = 0$ and $(\mathcal{S}, \mathcal{T})$ is minimal cut, then F is a maximal flow.
 C. If $F(\mathcal{S}, \mathcal{T}) = \text{capacity}(\mathcal{S}, \mathcal{T})$, then F is a maximal flow.
 D. If ab is an unsaturated edge with non-zero flow and $(\mathcal{S}, \mathcal{T})$ is a minimal cut, then either both vertices are in $\mathcal{S}$ or both vertices are in $\mathcal{T}$.

8. Use Inclusion-Exclusion for part A&B. (For those who don't know, dice are cube-shaped and thus have 6 faces. On each face there are from one to six dots, representing the numbers one to six. Each die has a face with each number between one and six.) [*Hint* : define the sets A_i so as to count the intersection of the complements of A_i.]
A&B. Count the number of ways of rolling 10 distinct dice so that at least one of each of the numbers 1-6 appears.
 C. Compute the probability that the above event occurs.

MAD 3105 Discrete Math 2 Section 1

Class Handout

---

1. M 8 Jan Homework 9.4 1 - 37 = 1 mod 4.
2. W 10 Jan Homework 9.4 3 - 35 = 3 mod 4. Quiz 1
3. F 12 Jan Homework 7.1 1 - 17 odd; 8.2 1, 4.

---

4. M 15 Jan No Class
5. W 17 Jan Homework 8.2 6, 7. Read 5.1
6. F 19 Jan Homework 5.1 1 - 19 odd, 18, 20 Quiz 2

---

7. M 22 Jan Homework 5.2 1 - 13 = 1 mod 4, 8.2 3, #11 on sheet
8. W 24 Jan Homework 5.3 1 - 4 and 5 - 17 = 1 mod 4. Quiz 3
9. F 26 Jan Homework 5.4 1 - 13 = 1 mod 4

---

10. M 29 Jan Review
11. W 31 Jan Test 1
12. F 2 Feb Homework 5.5 1 - 13 = 1 mod 3

---

13. M 5 Feb Homework 5.1 31; 5.5 15; 8.1 13 - 25 odd. Try the program named ~bellenot/match.tcl. Here are some pointers to tcl/tk, I recommend the book by Welch, prac prog in (with?) tcl/tk.
    - The Home Page
    - Another One
    - A Third
14. W 7 Feb Homework 8.1 29 - 35 odd Quiz 4
15. F 9 Feb Homework 8.3 1 - 25 = 1 mod 4

---

16. M 12 Feb Homework 8.3 3 - 27 = 3 mod 4
17. W 14 Feb Homework Like 8.1 31 and 35 but having no two consecutive zero's and not containing the pattern 01. Quiz 5
18. F 16 Feb Homework 8.2 13 - 25 = 1 mod 4.

---

19. M 19 Feb Homework (a)-(d) on Divide & Conquer
20. W 21 Feb Homework 8.4 1 - 25 = 1 mod 4; (e) on Divide and Conquer.
21. F 23 Feb Homework 1 - 4 on strong induction handout Quiz 6

---

22. M 26 Feb Homework 6.1 1 - 33 = 1 mod 4; #5 on strong induction handout
23. W 28 Feb Homework 6.2 1 - 29 = 1 mod 4; Quiz 7
24. F 1 Mar Homework 6.2 3 - 27 = 3 mod 4;

---

25. M 4 Mar Review
26. W 6 Mar Test 2
27. F 8 Mar Reduce to a problem you have already solved. Try it again, it might have been a fluke.

---

28. M 11 Mar Homework 6.3 1 - 21 = 1 mod 4;

29. W 13 Mar Homework 6.4 1 - 21 = 1 mod 4;Quiz 8
30. F 15 Mar Homework Big Oh handout 1, 5, 7, 11, 12

---

31. Spring Break

---

32. M 25 Mar Homework 7.2 1 - 33 = 1 mod 4;
33. W 27 Mar Homework 7.3 1 - 33 = 1 mod 4; Quiz 9
34. F 29 Mar Homework 7.4 1 - 33 = 1 mod 4;

---

35. M 1 Apr Homework 7.4 3 - 31 = 3 mod 4; 7.5 1 - 33 = 1 mod 4.
36. W 3 Apr Homework 7.5 3 - 31 = 3 mod 4; 7.6 1 - 21 = 1 mod 4. Quiz 10
37. F 5 Apr Homework 7.6 3 - 23 = 3 mod 4; 7.7 1 - 31 = 1 mod 4.

---

38. M 8 Apr Review
39. W 10 Apr Test 3
40. F 12 Apr Homework 8.5 1 - 25 = 1 mod 4.

---

41. M 15 Apr Homework 8.6 1 - 29 = 1 mod 4.
42. W 17 Apr Review Plan B
43. F 19 Apr Review

---

44. R 25 Apr Final -- 7:30 - 9:30 am

last update 18 Apr 96.