THE UNIVERSITY OF SYDNEY

MATH2069/2969 DISCRETE MATHEMATICS AND GRAPH THEORY

Semester 1 **Discrete Mathematics Tutorial 3 (Week 4)** 2006

Starred questions are suitable for students enrolled in MATH2969 or for students aiming for a credit or higher.

1. Let $X = 111110$ and $Y = 110111$ be integers written in base 2. Thus X corresponds to $2^5 + 2^4 + 2^3 + 2^2 + 2 = 62$ in base 10.

(a) To which integer does Y correspond in base 10?

(b) Find XY using traditional long multiplication in base 2 arithmetic. What does your answer correspond to in base 10?

(c) Observe that $X = 10^3 A + B$ and $Y = 10^3 B + A$ where $A = 111$ and $B = 110$. Find AB and $(A-B)^2$ working in base 2. Now find

$$10^6 AB + 10^3\big((A-B)^2 + 10AB\big) + AB\ .$$

(d) Can you explain why you get the same answer as (b)? (This is an example of the *divide-and-conquer* technique. The calculations AB and $(A-B)^2$ and the merged formula in (c) are intended to be briefer than (b).)

2. Let $P(x_1, y_1)$, $Q(x_2, y_2)$, $R(x_3, y_3)$ be points in the xy-plane, and consider the vectors $\mathbf{u} = \overrightarrow{PQ}$ and $\mathbf{v} = \overrightarrow{PR}$.

(a) Put

$$\Delta_{PQR} \quad = \quad \det \begin{bmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{bmatrix}$$

and verify that

$$\Delta_{PQR} \quad = \quad (x_2 - x_1)(y_3 - y_1) - (x_3 - x_1)(y_2 - y_1)\ .$$

(b) Find the cross product $\mathbf{u} \times \mathbf{v}$ where $\mathbf{u}$ and $\mathbf{v}$ are regarded as vectors in space, and compare your answer to part (a).

(c) Explain why the triangle $\triangle PQR$ is oriented anticlockwise if $\Delta_{PQR} > 0$ and clockwise if $\Delta_{PQR} < 0$. What happens if $\Delta_{PQR} = 0$?

(d) Let $P = (5, 1)$, $Q = (7, 9)$ and $R = (1, 4)$. What is Δ_{PQR}? In each case below, find Δ_{PQS}, Δ_{PSR} and Δ_{SQR} and deduce whether S lies inside or outside the triangle $\triangle PQR$:

(i) $S = (3, 3)$ (ii) $S = (4, 7)$ (iii) $S = (6, 5)$

3. Apply BUBBLESORT to the input sequence $2\,,\,6\,,\,4\,,\,8\,,\,1\,,\,7\,,\,5\,,\,3$. How many passes of the outer loop ("percolations") are needed to sort the sequence. List each sequence as it stands after each of these passes. How many swaps (in the inner loop) took place overall?

4. Consider the following reformulation of MERGESORT for sorting an input sequence $a_1, \ldots, a_N$ where $N = 2^k$ is a power of 2. The algorithm makes k passes then stops.

Pass 0: Write down $a_1, \ldots, a_N$.

Pass 1: Rewrite whilst sorting each successive pair of numbers.

Pass 2: Rewrite whilst merging each successive pair of pairs.

$\vdots$

Pass i: Rewrite whilst merging each successive pair of lists of 2^{i-1} elements.

$\vdots$

Pass k: Rewrite whilst merging the final pair of lists of 2^{k-1} elements.

(a) Apply this algorithm to the input sequence of the previous question.

*(b) In general, total up the number of times numbers are written down and an upper bound of comparisons made in the duration of this algorithm. Deduce that MERGESORT has $O(N \log N)$ running time.

*(c) How would you modify the algorithm and the calculation in (b) to deal with input sequences of any positive length?

5. (a) Verify that $\lceil X + M \rceil \;=\; \lceil X \rceil + M$ for any real number X and integer M.

*(b) Verify that if X is any positive real number and M any positive integer then

$$\left\lceil \frac{X}{M} \right\rceil \;=\; \left\lceil \frac{\lceil X \rceil}{M} \right\rceil .$$

Give an example to show this equation can fail if M is allowed to be real.

*(c) Fix a positive real X and a positive integer M. Define $a_0 = \lceil X \rceil$ and $a_{n+1} = \lceil a_n / M \rceil$ for $n \geq 0$. Prove that

$$a_n \;=\; \left\lceil \frac{X}{M^n} \right\rceil$$

for each n. What is $\lim_{n \to \infty} a_n$?

6. Suppose we have a strictly increasing sequence $\mathcal{S}$ of $N = 2^k$ real numbers: $a_1 < a_2 < \ldots < a_{N-1} < a_N$. Put

$$\mathcal{S}_L = \{a_1, \ldots, a_{N/2}\} \quad \text{and} \quad \mathcal{S}_R = \{a_{N/2+1}, \ldots, a_N\} ,$$

and write

$$M_L = M_L(\mathcal{S}) = a_{N/2} \; , \quad M_R = M_R(\mathcal{S}) = a_{N/2+1} \; .$$

The following recursive algorithm takes a real number as an input and locates it in a half-closed interval associated with $\mathcal{S}$:

BINARYSEARCH($\mathcal{S}$): Input a real number λ.

(1) If $\mathcal{S} = \{X\}$ output $(-\infty, X)$ if $\lambda < X$; output $[X, \infty)$ otherwise, and stop.

(2) If $M_L \le \lambda < M_R$ output $[M_L, M_R)$ and stop.

(3) If $\lambda < M_L$ then input λ to BINARYSEARCH($\mathcal{S}_L$).

(4) Otherwise input λ to BINARYSEARCH($\mathcal{S}_R$).

(a) Describe what happens when BINARYSEARCH $\big(1, 2, \ldots, 64\big)$ is applied to the inputs $\lambda = 32.4$, $\lambda = 12.97$, $\lambda = 35.1$ and $\lambda = -1$.

(b) Explain why the running time $T(N)$ of the algorithm, as a function of the number $N (= 2^k)$ of elements in the original ordered list, is described by

$$T(N) \;=\; T(N/2) + O(1) \; .$$

*(c) Solve the recurrence of (b) to show that $T(N) = O(\log N)$.

*(d) How would you modify the specification of BINARYSEARCH($\mathcal{S}$) where the size of $\mathcal{S}$ is any positive integer.

**(e) Relate the following recurrence to the running time of your modified algorithm in part (d):

$$T(N) \;=\; T(\lceil N/2 \rceil) + O(1) \; .$$

Solve this and show that you get the same conclusion as in part (c).