

1. (a) In base 10 we have  $Y = 32 + 16 + 4 + 2 + 1 = 55$ .  
 (b) Using long multiplication we get

$$\begin{array}{r}
 111110 \\
 \times 110111 \\
 \hline
 111110 \\
 1111100 \\
 11111000 \\
 1111100000 \\
 11111000000 \\
 \hline
 110101010010
 \end{array}$$

and in base 10 the answer is

$$2^{11} + 2^{10} + 2^8 + 2^6 + 2^4 + 2 = 3,410.$$

- (c) Observe that  $(A - B)^2 = 1$  and

$$AB = 1110 + 11100 = 101010$$

so  $10^6 AB + 10^3((A - B)^2 + 10AB) + AB$  becomes the following sum

$$\begin{array}{r}
 101010000000 \\
 1010101000 \\
 101010 \\
 \hline
 110101010010
 \end{array}$$

- (d) The answers must coincide because, in base 2,

$$\begin{aligned}
 (A - B)^2 + 10AB &= A^2 - AB - AB + B^2 + 10AB \\
 &= A^2 - 10AB + B^2 + 10AB \\
 &= A^2 + B^2,
 \end{aligned}$$

giving

$$\begin{aligned}
 10^6 AB + 10^3((A - B)^2 + 10AB) + AB &= 10^6 AB + 10^3(A^2 + B^2) + AB \\
 &= (10^3 A + B)(10^3 B + A) \\
 &= XY.
 \end{aligned}$$

2. (a) Using a couple of elementary column operations, we get

$$\begin{aligned}
 \Delta_{PQR} &= \det \begin{bmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{bmatrix} \\
 &= \det \begin{bmatrix} 1 & 0 & 0 \\ x_1 & x_2 - x_1 & x_3 - x_1 \\ y_1 & y_2 - y_1 & y_3 - y_1 \end{bmatrix} \\
 &= \det \begin{bmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \end{bmatrix} \\
 &= (x_2 - x_1)(y_3 - y_1) - (x_3 - x_1)(y_2 - y_1) .
 \end{aligned}$$

- (b) Regarded as vectors in space, and noting that  $\mathbf{i} \times \mathbf{i} = \mathbf{j} \times \mathbf{j} = \mathbf{0}$ ,  $\mathbf{i} \times \mathbf{j} = \mathbf{k}$  and  $\mathbf{j} \times \mathbf{i} = -\mathbf{k}$ , we get

$$\begin{aligned}
 \mathbf{u} \times \mathbf{v} &= \overrightarrow{PQ} \times \overrightarrow{PR} \\
 &= [(x_2 - x_1)\mathbf{i} + (y_2 - y_1)\mathbf{j}] \times [(x_3 - x_1)\mathbf{i} + (y_3 - y_1)\mathbf{j}] \\
 &= (x_2 - x_1)(y_3 - y_1)\mathbf{k} - (y_2 - y_1)(x_3 - x_1)\mathbf{k} \\
 &= [(x_2 - x_1)(y_3 - y_1) - (y_2 - y_1)(x_3 - x_1)]\mathbf{k} ,
 \end{aligned}$$

the coefficient of which coincides with  $\Delta_{PQR}$  calculated in (a).

- (c) If  $\Delta_{PQR} > 0$  then  $\mathbf{u} \times \mathbf{v}$  points in the direction of the positive  $z$ -axis, which, by the right-hand rule for cross-products, means that  $\mathbf{u}$  must be rotated anticlockwise to point in the same direction as  $\mathbf{v}$ , that is, the triangle  $\Delta PQR$  is oriented anticlockwise as one moves from  $P$  to  $Q$  to  $R$ . If  $\Delta_{PQR} < 0$  then  $\mathbf{u} \times \mathbf{v}$  points in the direction of the negative  $z$ -axis, which, by the right-hand rule, means that  $\mathbf{u}$  must be rotated clockwise to point in the same direction as  $\mathbf{v}$ , that is, the triangle  $\Delta PQR$  is oriented clockwise. If  $\Delta_{PQR} = 0$  then  $P, Q, R$  are collinear.

- (d) We have

$$\begin{aligned}
 \Delta_{PQR} &= \det \begin{bmatrix} 1 & 1 & 1 \\ 5 & 7 & 1 \\ 1 & 9 & 4 \end{bmatrix} = (28 - 9) - (20 - 1) + (45 - 7) \\
 &= 19 - 19 + 38 = 38 > 0
 \end{aligned}$$

- (i) For  $S = (3, 3)$  we have

$$\begin{aligned}
 \Delta_{PQS} &= \det \begin{bmatrix} 1 & 1 & 1 \\ 5 & 7 & 3 \\ 1 & 9 & 3 \end{bmatrix} = (21 - 27) - (15 - 3) + (45 - 7) \\
 &= -6 - 12 + 38 = 20 > 0
 \end{aligned}$$

$$\begin{aligned}
 \Delta_{PSR} &= \det \begin{bmatrix} 1 & 1 & 1 \\ 5 & 3 & 1 \\ 1 & 3 & 4 \end{bmatrix} = (12 - 3) - (20 - 1) + (15 - 3) \\
 &= 9 - 19 + 12 = 2 > 0
 \end{aligned}$$

$$\begin{aligned}\Delta_{SQR} &= \det \begin{bmatrix} 1 & 1 & 1 \\ 3 & 7 & 1 \\ 3 & 9 & 4 \end{bmatrix} = (28 - 9) - (12 - 3) + (27 - 21) \\ &= 19 - 9 + 6 = 16 > 0\end{aligned}$$

Since all orientations are anticlockwise  $S$  must lie inside  $\triangle PQR$ .

(ii) For  $S = (4, 7)$  we have

$$\begin{aligned}\Delta_{PQS} &= \det \begin{bmatrix} 1 & 1 & 1 \\ 5 & 7 & 4 \\ 1 & 9 & 7 \end{bmatrix} = (49 - 36) - (35 - 4) + (45 - 7) \\ &= 13 - 31 + 38 = 20 > 0\end{aligned}$$

$$\begin{aligned}\Delta_{PSR} &= \det \begin{bmatrix} 1 & 1 & 1 \\ 5 & 4 & 1 \\ 1 & 7 & 4 \end{bmatrix} = (16 - 7) - (20 - 1) + (35 - 4) \\ &= 9 - 19 + 31 = 21 > 0\end{aligned}$$

$$\begin{aligned}\Delta_{SQR} &= \det \begin{bmatrix} 1 & 1 & 1 \\ 4 & 7 & 1 \\ 7 & 9 & 4 \end{bmatrix} = (28 - 9) - (16 - 7) + (36 - 49) \\ &= 19 - 9 - 13 = -3 < 0\end{aligned}$$

Since orientations are mixed  $S$  must lie outside  $\triangle PQR$ .

(iii) For  $S = (6, 5)$  we have

$$\begin{aligned}\Delta_{PQS} &= \det \begin{bmatrix} 1 & 1 & 1 \\ 5 & 7 & 6 \\ 1 & 9 & 5 \end{bmatrix} = (35 - 54) - (25 - 6) + (45 - 7) \\ &= -19 - 19 + 38 = 0\end{aligned}$$

$$\begin{aligned}\Delta_{PSR} &= \det \begin{bmatrix} 1 & 1 & 1 \\ 5 & 6 & 1 \\ 1 & 5 & 4 \end{bmatrix} = (24 - 5) - (20 - 1) + (25 - 6) \\ &= 19 - 19 + 19 = 19 > 0\end{aligned}$$

$$\begin{aligned}\Delta_{SQR} &= \det \begin{bmatrix} 1 & 1 & 1 \\ 6 & 7 & 1 \\ 5 & 9 & 4 \end{bmatrix} = (28 - 9) - (24 - 5) + (54 - 35) \\ &= 19 - 19 + 19 = 19 > 0\end{aligned}$$

Since two orientations are anticlockwise and  $P, Q, S$  are collinear,  $S$  must lie on the boundary of  $\triangle PQR$  (in fact on the edge  $\overline{PQ}$ ).

3. Starting with 2, 6, 4, 8, 1, 7, 5, 3 we get the following percolations:

2, 4, 6, 1, 7, 5, 3, 8	(5 swaps)
2, 4, 1, 6, 5, 3, 7, 8	(3 swaps)
2, 1, 4, 5, 3, 6, 7, 8	(3 swaps)
1, 2, 4, 3, 5, 6, 7, 8	(2 swaps)
1, 2, 3, 4, 5, 6, 7, 8	(1 swap)

Five percolations are enough to sort the sequence using 14 swaps overall.

4. (a) The sequence from the previous question is manipulated as follows:

2, 6, 4, 8, 1, 7, 5, 3	2, 6		4, 8		1, 7		3, 5
2, 4, 6, 8		1, 3, 5, 7	1, 2, 3, 4, 5, 6, 7, 8				

\*(b) To merge two lists of size  $2^{i-1}$ , up to  $2^i - 1$  comparisons are made (because the last number is forced) and this is done  $N/2^i = 2^{k-i}$  times in the  $i$ th pass for  $1 \leq i \leq k$ . Adding to this  $N$  instances of writing down each number at each pass, the running time may be estimated to be

$$\begin{aligned}
 T(N) &= N + \sum_{i=1}^k (N + 2^{k-i}(2^i - 1)) = N + \sum_{i=1}^k (N + 2^k - 2^{k-i}) \\
 &= N + \sum_{i=1}^k O(2N) = N + (\log_2 N)O(N) = O(N \log N).
 \end{aligned}$$

\*(c) Suppose the input sequence has length any positive integer  $N$  which is not a power of 2. Then  $2^{k-1} < N < 2^k$  for some  $k$ . Note that  $2N > 2^k$ . We can modify the algorithm by adding a prior instruction to add  $2^k - N$  fillers of the form  $\infty, \dots, \infty$  to the sequence, where  $\infty$  behaves as being bigger than any number. Applying the previous algorithm now yields a sorted sequence which terminates in the padding  $\infty, \dots, \infty$  which can then be deleted. The running time is, from (b), but with an adjustment for the addition and deletion of padding,

$$\begin{aligned}
 T(N) &= O(2^k \log(2^k)) + O(2^k - N) + O(2^k - N) \\
 &= O(2N \log(2N)) + O(2N) + O(2N) \\
 &= O(N \log N + N \log 2) + O(N) = O(N \log N).
 \end{aligned}$$

5. (a) By definition  $\lceil X \rceil$  is the smallest integer  $\geq X$ . Certainly then  $\lceil X \rceil + M$  is an integer  $\geq X + M$ . But  $\lceil X \rceil - 1$  is an integer less than  $X$  so  $\lceil X \rceil + M - 1$  is an integer less than  $X + M$ . This proves that  $\lceil X \rceil + M$  is the smallest integer  $\geq X + M$ , whence  $\lceil X + M \rceil = \lceil X \rceil + M$ .

\*(b) If  $X$  is an integer then  $X = \lceil X \rceil$  and the result is obviously true. We therefore suppose  $X$  is not an integer and write

$$X = a_0 + a_1M + a_2M^2 + \dots + a_NM^N$$

for some nonnegative integers  $N, a_1, \dots, a_N$  and a real number  $a_0$  such that  $0 < a_0 < M$ . Observe that

$$0 < a_0 \leq \lceil a_0 \rceil \leq M,$$

so that

$$0 < a_0/M \leq \lceil a_0 \rceil / M \leq 1,$$

whence

$$\left\lceil \frac{a_0}{M} \right\rceil = \left\lceil \frac{\lceil a_0 \rceil}{M} \right\rceil = 1.$$

If  $X = a_0$  then we are done, so we suppose  $X \neq a_0$  giving  $N > 0$ . Then, using (a),

$$\begin{aligned} \left\lceil \frac{X}{M} \right\rceil &= \left\lceil \frac{a_0}{M} + a_1 + \dots + a_N M^{N-1} \right\rceil \\ &= \left\lceil \frac{a_0}{M} \right\rceil + a_1 + \dots + a_N M^{N-1} \\ &= \left\lceil \frac{\lceil a_0 \rceil}{M} \right\rceil + a_1 + \dots + a_N M^{N-1} \\ &= \left\lceil \frac{\lceil a_0 \rceil}{M} + a_1 + \dots + a_N M^{N-1} \right\rceil \\ &= \left\lceil \frac{\lceil a_0 \rceil + a_1 M + \dots + a_N M^N}{M} \right\rceil \\ &= \left\lceil \frac{\lceil a_0 + a_1 M + \dots + a_N M^N \rceil}{M} \right\rceil = \left\lceil \frac{\lceil X \rceil}{M} \right\rceil. \end{aligned}$$

This equation fails, for example if  $X = 1.1 = M$  for then

$$\left\lceil \frac{X}{M} \right\rceil = 1 < 2 = \left\lceil \frac{2}{1.1} \right\rceil = \left\lceil \frac{\lceil X \rceil}{M} \right\rceil.$$

\*(c) Observe that  $a_0 = \lceil X \rceil = \lceil X/M^0 \rceil$ , which starts an induction. Suppose  $k \geq 0$  and  $a_k = \lceil X/M^k \rceil$ . Then, using (b),

$$a_{k+1} = \lceil a_k/M \rceil = \left\lceil \frac{\lceil x/M^k \rceil}{M} \right\rceil = \left\lceil \frac{x/M^k}{M} \right\rceil = \left\lceil \frac{x}{M^{k+1}} \right\rceil,$$

which established the inductive step. The result now follows by induction. Clearly  $\lim_{n \rightarrow \infty} a_n = 1$ .

6. (a) Use B as an abbreviation for BINARYSEARCH.

If we input  $\lambda = 32.4$  then at step **(2)**,  $B(1, \dots, 64)$  halts and outputs the interval  $[32, 33)$ .

If we input  $\lambda = 12.97$  then  $B(1, \dots, 64)$  calls on  $B(1, \dots, 32)$  which in turn calls on  $B(1, \dots, 16)$ , which in turn calls on  $B(9, \dots, 16)$ , which halts at its step **(2)** and outputs the interval  $[12, 13)$ .

If we input  $\lambda = 35.1$  then  $B(1, \dots, 64)$  calls on  $B(33, \dots, 64)$  which in turn calls on  $B(33, \dots, 48)$ , which in turn calls on  $B(33, \dots, 40)$ , which in turn calls on  $B(33, \dots, 36)$ , which in turn calls on  $B(35, 36)$ , which halts at its step **(2)** and outputs the interval  $[35, 36)$ .

If we input  $\lambda = -1$  then  $B(1, \dots, 64)$  calls on  $B(1, \dots, 32)$  which in turn calls on  $B(1, \dots, 16)$ , which in turn calls on  $B(1, \dots, 8)$ , which in turn calls on  $B(1, \dots, 4)$ , which in turn calls on  $B(1, 2)$ , which in turn calls on  $B(1)$ , which halts at its step **(1)** and outputs the interval  $(-\infty, 1)$ .

(b) Steps **(1)** and **(2)** collectively take  $O(1)$  time. Only one of steps **(3)** or **(4)** is executed which use  $T(N/2)$  time as the sequences  $\mathcal{S}_L$  and  $\mathcal{S}_R$  have length  $N/2$ . Adding these gives

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

\*(c) Observe that

$$\begin{aligned} T(N) &= T(N/2) + O(1) = T(N/4) + 2O(1) \\ &= T(N/8) + 3O(1) = \dots = T(N/2^k) + kO(1) \\ &= T(1) + (\log_2 N)O(1) = O(1) + O(\log N) \\ &= O(\log N). \end{aligned}$$

\*(d) The instructions remain the same for any  $N$ . However we adjust the definitions of  $\mathcal{S}_L$ ,  $\mathcal{S}_R$ ,  $M_L$  and  $M_R$  as follows:

$$\begin{aligned} \mathcal{S}_L &= \{a_1, \dots, a_{\lceil N/2 \rceil}\}, \quad \mathcal{S}_R = \{a_{\lfloor N/2 \rfloor + 1}, \dots, a_N\}, \\ M_L &= a_{\lceil N/2 \rceil}, \quad M_R = a_{\lfloor N/2 \rfloor + 1}. \end{aligned}$$

Note that step **(2)** can only be executed at the moment  $B(\mathcal{S})$  is called if  $\mathcal{S}$  has an even number of elements, since if the size of  $\mathcal{S}$  is odd then  $M_L = M_R$ .

\*\* (e) The derivation of the recurrence is the same as in part (b) except that at steps **(3)** and **(4)** the new sequences  $\mathcal{S}_L$  and  $\mathcal{S}_R$  both have size  $\lceil N/2 \rceil$ . Consider now any positive integer  $N$ , so

$$2^{k-1} < N \leq 2^k.$$

for some nonnegative  $k$ . Note that  $2^k < 2N$  so  $k < \log_2 N + 1$ , giving  $k = O(\log N)$ . Iterating the recurrence, and using part (b) of the previous question with  $M = 2$ , we get

$$\begin{aligned}
 T(N) &= T(\lceil N/2 \rceil) + O(1) \\
 &= T\left(\left\lceil \frac{\lceil N/2 \rceil}{2} \right\rceil\right) + 2O(1) \\
 &= T\left(\left\lceil \frac{N}{4} \right\rceil\right) + 2O(1) \\
 &= T\left(\left\lceil \frac{\lceil N/4 \rceil}{2} \right\rceil\right) + 3O(1) \\
 &= T\left(\left\lceil \frac{N}{8} \right\rceil\right) + 3O(1) \\
 &= \dots \\
 &= T\left(\left\lceil \frac{N}{2^k} \right\rceil\right) + kO(1) \\
 &= T(1) + kO(1) \\
 &= O(1) + O(k) \\
 &= O(\log N) .
 \end{aligned}$$