McGill University COMP251: Assignment 1 Solution

Question 1 (a) **Idea** Let $k = \lceil \frac{n}{2} \rceil$. Then $A[k]$ is the median of A, and $B[k]$ is the median of B. The Divide-and-Conquer algorithm arises from the following observation:

Observation Suppose that $A(k) \geq B(k)$. Then the median of A and B is the same as the median of $A[1 \ \dots \ k]$ and $B[(\lfloor n/2 \rfloor + 1) \ \dots \ n]$.

To see why this is true, note that:

- Each element in $A[(k+1) \ \dots n]$ is not smaller than $2k$ elements

$$B[1 \ \dots \ k], A[1 \ \dots k]$$

- Similarly, each element in $B[1 \ \dots \ \lfloor n/2 \rfloor]$ is not larger than $2k$ elements

$$B[(\lfloor n/2 \rfloor + 1) \ \dots \ n], A[(k+1) \dots \ n]$$

Consequently, the median of A and B must be among

$$A[1 \ \dots \ k], B[(\lfloor n/2 \rfloor + 1) \ \dots \ n]$$

Moreover, *it is the median of the numbers in these two subarrays.*

Thus, in general we will find the median of the union of $A[x \ \dots \ (x+\ell-1)]$ and $B[y \ \dots \ (y+\ell-1)]$ (i.e. there are the same number of elements from A and B). The following program Median(x, y, ℓ) will do this. To solve the given problem, call Median$(1, 1, n)$.

Median(x, y, ℓ)

1. if $\ell = 1$ return $\min\{A[x], B[y]\}$
2. else
3. $\quad k \leftarrow \lceil \ell/2 \rceil$
4. $\quad$ if $A[x+k-1] \geq B[y+k-1]$
5. $\quad\quad$ return Median$(x, y + \lfloor \ell/2 \rfloor, \ell - k)$
6. $\quad$ else
7. $\quad\quad$ return Median$(x + \lfloor \ell/2 \rfloor, y, \ell - k)$
8. $\quad$ end if
9. end if

For each pair of subarrays of length ℓ, the program makes a recursive call to subarrays of length $\ell/2$. So the running time is $\log_2 n$.

(b) **Proof of Correctness** We prove by strong induction on $\ell \geq 1$ that Median(x, y, ℓ) returns the median of $A[x \ \dots \ (x+\ell-1)]$ and $B[y \ \dots \ (y+\ell-1)]$.

Base case: $\ell = 1$. By line 1 Median$(x, y, 1)$ returns $\min\{A[x], B[y]\}$ which is the median of $A[x], B[y]$.

Induction step: Assume Induction Hypothesis: for all x, y, ℓ' wher $\ell' < \ell$, Median(x, y, ℓ') returns the median of $A[x \ \dots \ (x+\ell'-1)]$ and $B[y \ \dots \ (y+\ell'-1)]$.

Consider Median(x, y, ℓ). Suppose that $A[x+k-1] \geq B[y+k-1]$. Then the output of Median(x, y, ℓ) is Median(x, y', ℓ') where $\ell' = \ell - k$ and $y' = y + \lfloor \ell/2 \rfloor$. By the Observation above, the median of $A[x \ \ldots \ (x+\ell-1)]$ and $B[y \ \ldots \ (y+\ell-1)]$ is the same as the median of $A[x \ \ldots \ (x+\ell'-1)]$ and $A[y' \ \ldots \ (y'+\ell'-1)]$. By the IH Median$(x, y', \ell')$ return the the latter median (we can apply the IH because $\ell' < \ell$), which is the correct answer.

Now suppose that $A[x+k-1] < B[y+k-1]$, then by the same argument, the median of $A[x \ \ldots \ (x+\ell-1)]$ and $B[y \ \ldots \ (y+\ell-1)]$ is the same as the median of $A[x' \ \ldots \ (x'+\ell'-1)]$ and $A[y \ \ldots \ (y+\ell'-1)]$, where $\ell' = \ell - k$ and $x' = x + \lfloor \ell/2 \rfloor$. Again, apply the IH for $\ell < \ell$.

This shows that for any x, y, Median(x, y, ℓ) returns the median of $A[x \ \ldots \ (x+\ell-1)]$ and $B[y \ \ldots \ (y+\ell-1)]$. So we are done.

Question 2 (a) We prove by strong induction on $j - i$ (roughly the length of $A[i \ \ldots j]$) that ZZZ-sort(A, i, j) sorts the subarray $A[i \ \ldots j]$ into non-decreasing order.

There are 2 base cases : $j - i = 0$ or $j - i = 1$ (i.e., the subarray has sizes 1 or 2). In both cases, the algorithm terminates at line 3. In the first case, $A[i \ \ldots j]$ is obviously sorted since it only contains 1 element. In the second case, lines 1 and 2 make sure that $A[i \ \ldots j]$ is sorted.

For the induction step, let $j \geq 2$ and assume the

Induction Hypothesis: For any $i, j \leq n$ such that $j - i < \ell$, ZZZ-sort(A, i, j) sorts the subarray $A[i \ \ldots j]$ into non-decreasing order.

Now, let i, j be such that $j - i = \ell$. We show that ZZZ-sort(A, i, j) sorts the subarray $A[i \ \ldots j]$ into non-decreasing order. By the induction hypothesis, line 5 sorts the last two thirds of $A[i \ \ldots j]$ (we can apply the IH because $j - (i + k) = \ell - k < \ell$). Similarly, line 6 sorts the first two thirds of the resulted subarray. As a result, the first third of the subarray contains the smallest elements of the subarray. Next, also by the IH, line 7 sorts the remaining two thirds. As a result the whole subarray is sorted. QED.

(b) The following recurrence describes the worst-case running time of ZZZ-sort :

$$T(n) = 3 * T(\frac{2 * n}{3}) + c$$

$$T(1) = c_1 \ ; \ T(2) = c_2$$

where c, c_1 and c_2 are constants that describe the running time of various operations such as those on lines 1 to 4.

Recall the Master Theorem. Applying it to our recurrence yields :

$$a = 3, \quad b = \frac{3}{2} \quad d = 0$$

Here $a > b^d$, so

$$T(n) = \Theta(n^{log_b a}) = \Theta(n^{log_{\frac{3}{2}} 3}) = \Theta(n^{2.7})$$