Họ tên: Nguyễn Trung Dũng

MSSV: 19120486

Bài 1:

a)

- **Phép so sánh là thao tác cơ sở**

⇒ Hệ thức try hồi là:

$$C(n) = \begin{cases} C\left(\left\lfloor\frac{n}{2}\right\rfloor\right) + C\left(n - \left\lfloor\frac{n}{2}\right\rfloor\right) + 2, & \text{n} > 2 \\ 1 \quad , & \text{n} \leq 2 \end{cases}$$

Giả sử $n = 2^k$

$$\Rightarrow C(n) = \begin{cases} 2C\left(\frac{n}{2}\right) + 2, & \text{n} > 2 \\ 1 \quad , & \text{n} \leq 2 \end{cases}$$

⇒ Ta có:

$$\begin{aligned} C(n) &= 2C\left(\frac{n}{2}\right) + 2 \\ &= 2\left(2C\left(\frac{n}{4}\right) + 2\right) + 2 \\ &= \cdots \\ &= 2^{k-1}C(2) + 2 + 4 + 8 + \cdots + 2^{k-1} \\ &= 2^{k-1} + 2\frac{1 - 2^{k-1}}{1 - 2} \\ &= 2^k + 2^{k-1} - 2 \approx n \in \Theta(n) \end{aligned}$$

- **Phép so sánh `if(l >= r-1)` cũng là thao tác cơ sở**

⇒ Hệ thức try hồi là:

$$C(n) = \begin{cases} C\left(\left\lfloor\frac{n}{2}\right\rfloor\right) + C\left(n - \left\lfloor\frac{n}{2}\right\rfloor\right) + 3, & \text{n} > 2 \\ 2 \quad , & \text{n} \leq 2 \end{cases}$$

Giả sử $n = 2^k$, ta có:

$$C(n) = 2C\left(\frac{n}{2}\right) + 3$$

$$= 2\left(2C\left(\frac{n}{4}\right) + 3\right) + 3$$

$$= 2^{k-1}C(2) + 3 + 6 + 12 + \cdots + 3.2^{k-2}$$

$$= 2^{k-1} + 3\frac{1 - 2^{k-1}}{1 - 2}$$

$$= 2.2^k - 3 \approx n \in \Theta(n)$$

b) Tiếp cận có cùng độ phức tạp với chia để trị nhưng không dùng đệ quy:

```
MinMax(a[1..n]) {
    min = max = a[1];
    for (i = 2; i ≤ n; i++) {
        if (a[i] > max)
            max = a[i];
        if (a[i] < min)
            min = a[i];
    }
    return <min, max>
}
```

- Các thao tác cơ sở là các phép so sánh
- Có i chạy từ 2 $\rightarrow n \Rightarrow$ tổng số lần thực hiện thao tác cơ sở: $2n - 2 \in \Theta(n)$

Bài 2:

a) Phép so sánh là thao tác cơ sở

- Trường hợp tốt nhất:

$$T(n) = \begin{cases} T\left(\left\lfloor\frac{n}{2}\right\rfloor\right) + T\left(\left\lceil\frac{n}{2}\right\rceil\right) + \left\lfloor\frac{n}{2}\right\rfloor, & \text{n} > 1 \\ 0 \quad , & \text{n} = 1 \end{cases}$$

Giả sử $n = 2^k$, ta có:

$$T(n) = 2T\left(\frac{n}{2}\right) + \frac{n}{2}$$

$$= 2\left(2T\left(\frac{n}{4}\right) + \frac{n}{4}\right) + \frac{n}{2}$$

$$= 2^kT(1) + \frac{n}{2} + \frac{n}{2} + \cdots + \frac{n}{2}(\text{k lần})$$

$$= k\frac{n}{2} = \frac{n}{2}.log_2n \in \Theta(nlogn)$$

- Trường hợp xấu nhất:

$$T(n) = \begin{cases} T\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + T\left(\left\lceil \frac{n}{2} \right\rceil\right) + (n-1), & \text{n} > 1 \\ 0 \qquad\qquad\qquad\qquad , & \text{n} = 1 \end{cases}$$

Giả sử $n = 2^k$, ta có:

$$\begin{aligned} T(n) &= 2T\left(\frac{n}{2}\right) + (n-1) \\ &= 2\left(2T\left(\frac{n}{4}\right) + \left(\frac{n}{2} - 1\right)\right) + (n-1) \\ &= 2^k T(1) + kn - (1 + 2 + 4 + \cdots + 2^{k-1}) \\ &= kn - \frac{1-2^k}{1-2} = nlog_2 n - n + 1 \in \Theta(nlogn) \end{aligned}$$

b) Phép chuyển dời là thao tác cơ sở

⇒ Hệ thức truy hồi:

$$M(n) = \begin{cases} M\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + M\left(\left\lceil \frac{n}{2} \right\rceil\right) + n, & \text{n} > 1 \\ 0 \qquad\qquad\qquad\qquad , & \text{n} = 1 \end{cases}$$

Giả sử $n = 2^k$, ta có:

$$\begin{aligned} M(n) &= 2M\left(\frac{n}{2}\right) + n \\ &= 2\left(2M\left(\frac{n}{4}\right) + \frac{n}{2}\right) + n \\ &= 2^k M(1) + kn \\ &= nlog_2 n \in \Theta(nlogn) \end{aligned}$$

Bài 3:

a) Phân bố dữ liệu thỏa điều kiện không giảm dần

b) Trong mã nguồn của hàm mergeSort(), thêm vòng for chạy từ low+1 tới high kiểm tra xem dữ liệu đã thỏa điều kiện như vậy hay chưa

```
mergeSort(a[1 .. n], low, high) {
    if (low < high) {
        for (i = low+1; i <= high; i++) {
            if (a[i] < a[i-1])
                break;
            if (i == high)
                return;
        }
        mid = ⌊(low + high) / 2⌋;
        mergeSort(a, low, mid);
        mergeSort(a, mid + 1, high);
        merge(a, low, mid, high);
    }
}
```