دانشکده علوم ریاضی و آمار

مدرس: دکتر مجتبی رفیعی نیمسال اول ۱۴۰۰-۱۴۰۱

ساختمان داده‌ها و الگوریتم‌ها

جلسه ۱۶

نگارنده: زهرا درویشی

۱۶ آبان ۱۴۰۰

فهرست مطالب

۱ الگوریتم مرتب سازی ادغامی

شبه کد مربوط به الگوریتم مرتب سازی ادغامی که حاوی دو تابع Merge-Sort و Merge است، در ادامه آورده شده است.

Algorithm 1 Merge-Sort($A[1 \cdots n], p, r$)

if $(p < r)$ **then**
 $q \leftarrow \left\lfloor \frac{(p+r)}{2} \right\rfloor$
 Merge-Sort $(A[1 \cdots n], p, q)$
 Merge-Sort $(A[1 \cdots n], q+1, r)$
 Merge $(A[1 \cdots n], p, q, r)$

Algorithm 2 Merge($A[1 \cdots n], p, q, r$)

```
1: ▷ Assume that A[p···q] and A[q+1···r] are sorted.
2: n1 ← q − p + 1
3: n2 ← r − q
4: Let L[1···n1+1] and R[1···n2+1] be two empty arrays.
5: for i = 1 to n1 do
6:     L[i] ← A[p+i−1]
7: for i = 1 to n2 do
8:     R[i] ← A[q+i]
9: L[n1+1] ← ∞,   R[n2+1] ← ∞
10: i ← 1,   j ← 1
11: for k = p to r do
12:     if (L[i] ≤ R[j]) then
13:         A[k] ← L[i]
14:         i ← i+1
15:     else
16:         A[k] ← R[j]
17:         j ← j+1
```

۱.۱ اثبات درستی الگوریتم

اثبات الگوریتم، شامل دو قسمت است:

- اثبات درستی الگوریتم Merge، توسط حلقه ناوردایی مربوط به خط‌های 10 تا 16 انجام می شود. در این راستا، خصیصه ناوردایی را به این صورت تعریف می‌کنیم: زیر آرایه $A[p...k-1]$ شامل $k-p$ کوچکترین عنصر آرایه‌های $L[1...n_1+1],\ R[1...n_2+1]$ به صورت مرتب است و همچنین $L[i],\ R[j]$ کوچکترین عناصر آرایه‌های خودشان هستند که در A کپی نشده‌اند. به راحتی می‌توانید گام‌های ناوردایی حلقه را برای خصیصه ذکر شده دنبال کنید. از اینرو ما از ذکر دقیق آنها خودداری کرده‌ایم.

- برای کل الگوریتم نیز از اثبات استقرایی استفاده می‌کنیم.
  - **پایه استقرا:** آرایه تک عنصری، یک آرایه مرتب است.
  - **گذار استقرا:** هر دو زیر آرایه با اندازه $\frac{n}{2}$ مرتب است و الگوریتم Marge هم نشان دادیم صحیح است، پس آرایه با اندازه n مرتب است.

۲.۱ تحلیل پیچیدگی الگوریتم

طبق فرم کلی بیان شده برای الگوریتم‌های تقسیم و غلبه:

$$T(n) = aT(\tfrac{n}{b}) + D(n) + C(n)$$
$$T(1) = c$$

داریم:

$$a = 2,\ \ b = 2,\ \ D(n) = \mathcal{O}(1),\ \ C(n) = \mathcal{O}(n)$$

بنابراین:

$$T(n) = 2T(\frac{n}{2}) + \mathcal{O}(n) + \mathcal{O}(1)$$

که $\mathcal{O}(1)$ جذب $\mathcal{O}(n)$ می‌شود و در نهایت داریم:

$$T(n) = 2T(\frac{n}{2}) + \mathcal{O}(n)$$