



نیمسال اول ۱۴۰۰-۱۴۰۱

مدرس: دکتر مجتبی رفیعی

## ساختمان داده‌ها و الگوریتم‌ها

جلسه ۱۵: مرتب‌سازی ادغامی

نگارنده: ملیکا زرنگار

۱۶ آبان ۱۴۰۰

### فهرست مطالب

- ۱ تابع پیچیدگی زمانی الگوریتم‌های تقسیم و غلبه
- ۲ مرتب‌سازی ادغامی
- ۳ ۱.۲ ادغام دو آرایه مرتب . . . . .

### ۱ تابع پیچیدگی زمانی الگوریتم‌های تقسیم و غلبه

فرم کلی برای تابع پیچیدگی زمانی الگوریتم‌های بازگشتی به صورت زیر می‌باشد:

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

جائیکه:

- $D(n)$ : زمان اجرای مرحله تقسیم برای  $n$  عنصر است،
- $C(n)$ : زمان اجرای مرحله ترکیب برای  $n$  عنصر است،
- ضریب  $a$ : بیانگر تعداد زیر مساله‌ها است که باید حل شود،

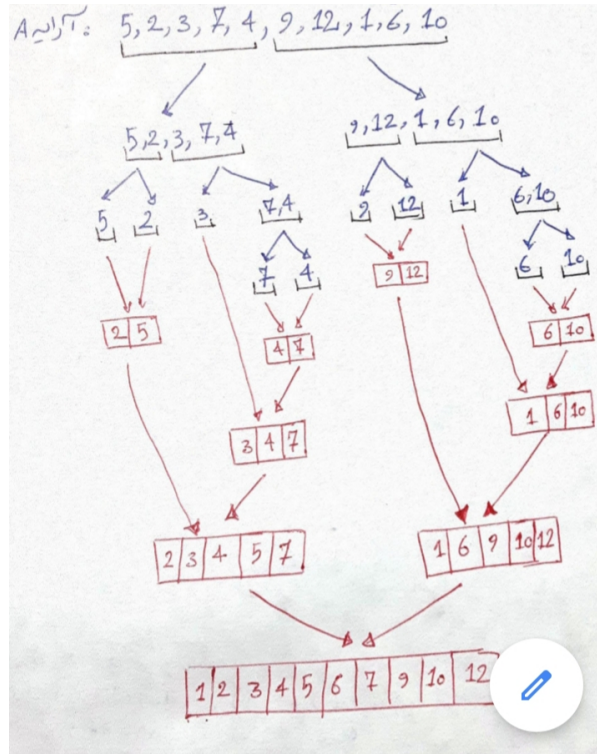
- ضریب  $1/b$  : بیانگر آن است که اندازه‌ی زیرمساله،  $1/b$  اندازه‌ی مساله اصلی است.

با توجه به مطالب فوق، تحلیل پیچیدگی الگوریتم مرتب‌سازی درجی-نسخه‌ی بازگشتی به صورت زیر می‌باشد:

$$T(n) = aT\left(\frac{n}{b}\right) + D(n) + C(n) \Rightarrow T(n) = T(n-1) + O(n) \Rightarrow T(n) = n * O(n) = O(n^2)$$

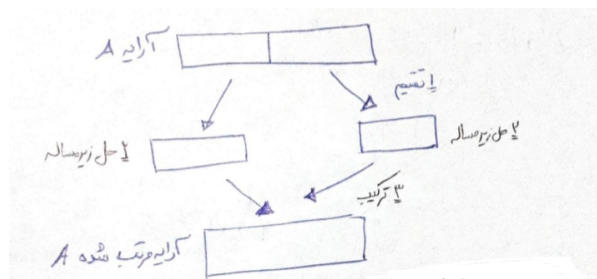
## ۲ مرتب‌سازی ادغامی

الگوریتمی است که از رویکرد تقسیم و حل برای مرتب‌سازی استفاده می‌کند. در ادامه الگوریتم مرتب‌سازی ادغامی با یک مثال تشریح می‌شود.



شکل ۱: نحوه عملکرد مرتب‌سازی ادغامی

شکل زیر روال کلی الگوریتم مرتب‌سازی ادغامی را به تصویر کشیده است.



شکل ۲: چارچوب کلی گام‌های تقسیم و غلبه برای مرتب‌سازی ادغامی

## ۱.۲ ادغام دو آرایه مرتب

با داشتن دو زیر آرایه مرتب  $A_1$  و  $A_2$  که به ترتیب هر یک شامل  $n_1$  و  $n_2$  عنصر هستند، می‌توان آنها را در زمان خطی  $O(n_1 + n_2)$  مرتب کرد. به طور دقیق‌تر داریم:

- حداقل زمان برابر  $n_1$  است، زمانی که تمامی عناصر  $A_1$  کوچکتر از  $A_2$  باشد.

- حداکثر زمان برابر  $n_1 + n_2 - 1$  است، زمانی که عناصر  $A_1$  و  $A_2$  یکی در میان از هم کوچکتر/بزرگتر باشند.

سوال: برای ادغام دو زیر آرایه  $A_1$  و  $A_2$  از آرایه  $A$ ، آیا نیاز به در نظر گرفتن آرایه اضافی داریم یا خیر؟ تحقیقات تا زمان نگارش سند نشان می‌دهد که تلاش‌های انجام شده برای این امر منجر به افزایش پیچیدگی زمانی این الگوریتم می‌شود.