



تمرینات سری 2

تمرینات تحلیل زمان بازگشتی:

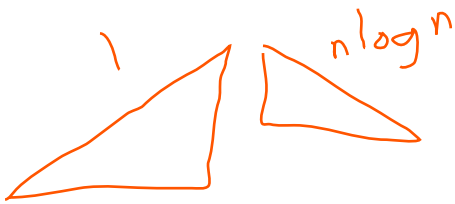
1- زمان اجرای الگوریتم‌های بازگشتی زیر را محاسبه نمایید.

- a. $T(n) = 2T\left(\frac{n}{2}\right) + n^4$
- b. $T(n) = 7T\left(\frac{n}{10}\right) + n$
- c. $T(n) = 7T\left(\frac{n}{3}\right) + n^2$
- d. $T(n) = \sqrt[4]{2}T\left(\frac{n}{4}\right) + \sqrt[5]{n}$
- e. $T(n) = 2T(n-2) + n^2$
- f. $T(n) = T\left(\frac{n}{2}\right) + T\left(\frac{n}{4}\right) + T\left(\frac{n}{8}\right) + n$
- g. $T(n) = 3T\left(\frac{n}{3} - 2\right) + \frac{n}{2}$
- h. $T(n) = 3T\left(\frac{n}{3}\right) + \frac{n}{\lg n}$
- i. $T(n) = 4T\left(\frac{n}{3}\right) + n \lg n$
- j. $T(n) = 4T\left(\frac{n}{2}\right) + n^2 \sqrt{n}$
- k. $T(n) = 2T(\sqrt{n}) + O(1)$
- l. $T(n) = T(\sqrt{n}) + O(\log \log n)$

تمرینات مرتب‌سازی:

2- مسئله زیر را در نظر بگیرید:

لیستی از فارغ‌التحصیلان کارآفرین دانشگاه صنعتی شاهرود که بر اساس سال ورود (و سپس نام) مرتب شده است، در اختیار داریم. می‌خواهیم این لیست را بر اساس شماره دانشجویی مرتب نماییم. دقت کنید که ابتدای تمام شماره‌های دانشجویی، سال ورود است.



شما برای حل این مسئله مرتب‌سازی درجی را برمی‌گزینید یا سریع؟ چرا؟

3- زمان اجرای مرتب‌سازی هرمی (heap sort) برای یک آرایه به طول n که به صورت صعودی مرتب شده است، چیست؟ برای حالت نزولی چطور؟

4- الگوریتم bucket sort را بر روی مقادیر زیر اجرا کرده و مراحل را به طور کامل نمایش دهید.

{0.79, 0.13, 0.16, 0.64, 0.39, 0.20, 0.89, 0.53, 0.71, 0.42}

5- به سوالات زیر در رابطه با مرتب‌سازی سریع (Quick Sort) پاسخ دهید:

یعنی آرایه های حاصل

باشند $n-1$ دارای طولهای 1 و

ا. رابطه بازگشتی الگوریتم چیست و مرتبه اجرایی آن چیست؟

best = $T(n) = 2T(n/2) + \text{teta}(n) = O(n \log n)$
worst = $T(n-1) + \text{teta}(n) = O(n^2)$

ب. چه زمانی این الگوریتم بدترین عملکرد را خواهد داشت؟

ج. مزیت این الگوریتم نسبت به مرتب سازی ادغامی چیست؟

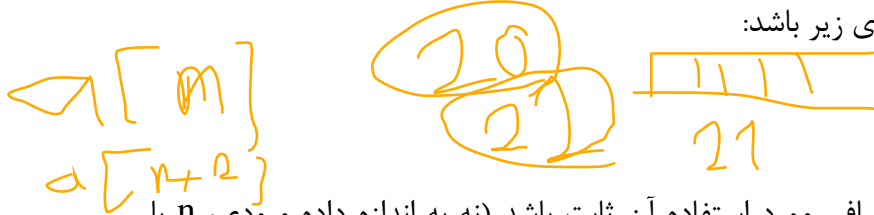
د. الگوریتم را روی آرایه زیر انجام داده و مراحل را نمایش دهید.

تونی

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

6- فرض کنید آرایه‌ای از مقادیر 0 و 1 به طول n دارید. می‌خواهید این آرایه را مرتب نمایید. یک الگوریتم

مرتب‌سازی ممکن است دارای ویژگیهای زیر باشد:



1- در زمان $O(n)$ اجرا شود.

2- پایدار باشد.

3- درجا باشد، یعنی اندازه فضای اضافی مورد استفاده آن ثابت باشد (نه به اندازه داده ورودی، n یا

توابعی از آن).

ا. الگوریتمی را نام ببرید که دارای ویژگیهای 1 و 2 باشد.

ب. الگوریتمی را نام ببرید که دارای ویژگیهای 1 و 3 باشد.

ج. الگوریتمی را نام ببرید که دارای ویژگیهای 2 و 3 باشد.

د. آیا هیچکدام از الگوریتم‌هایی که در مراحل الف تا ج توصیف کردید، قادر است n داده b بیتی را در

ترکیب با radix sort در زمان $O(bn)$ مرتب نماید؟ اگر بله، چگونه و اگر خیر، چرا؟

Insertion
insertion
insertion

7- فرض کنید آرایه‌ای از مقادیر 1 تا k به طول n دارید. توضیح دهید چگونه می‌توان مرتب‌سازی شمارشی (counting sort) را تغییر داد تا داده‌ها به صورت درجا و در زمان $O(n + k)$ مرتب شوند. البته مجاز هستید که برای شمارش، فضایی به اندازه k را مصرف نمایید.

8- آرایه مرتب‌شده‌ای از n عدد داریم. 10 بار، هر بار دو عنصر از آرایه را به صورت رندم جابجا می‌کنیم. ثابت کنید می‌توان آرایه را در $O(n)$ مرتب کرد.