

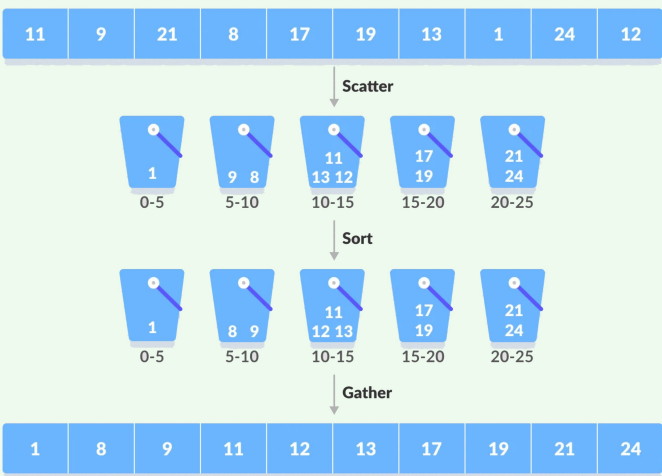


در جلسه گذشته به ادامه معرفی روش های مرتب‌سازی خطی پرداختیم و دو روش مرتب‌سازی سطلی و مرتب‌سازی مبنایی را معرفی کردیم. سپس مساله انتخاب را مورد بررسی قرار دادیم. ابتدا روش‌های مرتب‌سازی خطی را بررسی می‌کنیم:

- **مرتب سازی سطلی:** در این مرتب‌سازی عناصر را با استفاده از یک تابع درهم‌سازی مرتب، داخل تعدادی سطل قرار می‌دهیم. سپس عناصر درون هر سطل را با استفاده از روش مرتب‌سازی دلخواه مرتب کرده و عناصر مرتب‌شده سطل‌های مختلف را به ترتیب پشت سر هم قرار می‌دهیم. تابع درهم‌سازی مرتب به صورت زیر تعریف می‌شود:

تابع f یک تابع درهم‌سازی مرتب است $\Leftrightarrow x < y \rightarrow f(x) \leq f(y)$

در شکل زیر مثالی از مرتب‌سازی سطلی آمده است:



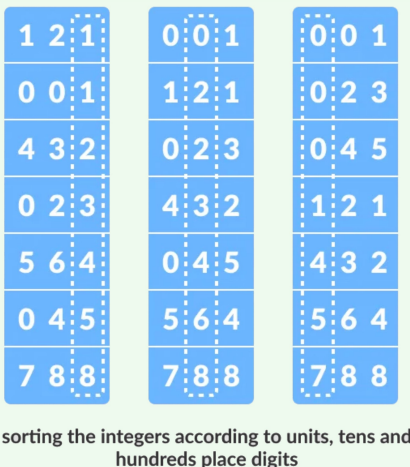
حال مرتبه زمانی این الگوریتم را مورد بررسی قرار می‌دهیم. فرض می‌کنیم که عناصر درون سطل به کمک یک الگوریتم مرتب‌سازی (مثلا مرتب‌سازی درجی) که در بدترین حالت از مرتبه $O(n^2)$ است، مرتب شده است.

- بدترین حالت : در این حالت تمامی عناصر در یک سطل قرار گرفته و الگوریتم از مرتبه $O(n^2)$ خواهد بود.
- حالت متوسط: با فرض یکنواختی، اگر m سطل داشته باشیم؛ درون هر سطل $\frac{m}{n}$ عنصر قرار می‌گیرد و در نتیجه الگوریتم از مرتبه $O(\frac{n^2}{m} + n)$ می‌باشد.

- **مرتب سازی مبنایی:** فرض می‌کنیم کلیدهای ورودی اعداد حداکثر d رقمی هستند. از کم‌ارزش‌ترین رقم طبق الگوریتم زیر شروع می‌کنیم:

{ عناصر آرایه A را بر حسب رقم i با استفاده از یک روش مرتب سازی پایدار مرتب می کنیم } $for(i : 1 \rightarrow d)$

اگر در الگوریتم بالا از روش مرتب‌سازی شمارشی به عنوان مرتب‌سازی پایدار استفاده کنیم، الگوریتم از مرتبه $O(nd)$ خواهد بود. در شکل زیر مثالی از مرتب‌سازی مبنایی آمده است:



مساله انتخاب : آرایه A شامل n عدد داده شده است. می‌خواهیم عنصر مرتبه kام آرایه (عنصری که در حالت مرتبه شده در جایگاه kام قرار می‌گیرد) را انتخاب کنیم:

الگوریتم q-select:

۱. آرایه را بر حسب عنصر دلخواه x پارتیشن می‌کنیم (p محل عنصر x است).
 ۲. اگر $p = k$: عنصر x را برمی‌گردانیم.
 ۳. اگر $p > k$: به طور بازگشتی عنصر k ام را در $A[1, 2, ..., p-1]$ پیدا می‌کنیم.
 ۴. اگر $p < k$: به طور بازگشتی عنصر $k-p$ ام را در آرایه $A[p+1, ..., n]$ پیدا می‌کنیم.
- در بدترین حالت هر بار عنصر محور، عنصر آخر آرایه می‌باشد؛ پس در این حالت الگوریتم از مرتبه $O(n^2)$ و در حالت متوسط الگوریتم از مرتبه $O(n)$ می‌باشد.

الگوریتم f-select: $(T(n))$

۱. عناصر آرایه A را به $\frac{n}{5}$ دسته با اندازه ۵ تقسیم می‌کنیم. $(O(n))$
 ۲. تمام دسته های ۵ تایی را مرتب می‌کنیم. $(O(n))$
 ۳. آرایه B شامل عناصر میانی هر دسته را تشکیل می‌دهیم. $(O(n))$
 ۴. عنصر میانه B را با استفاده از الگوریتم f-select به طور بازگشتی پیدا می‌کنیم. $(T(\frac{n}{5}))$
- فرض کنید x عنصر میانه و p محل x می‌باشد.
۵. آرایه A را بر حسب عنصر x ، پارتیشن می‌کنیم. $(O(n))$
- اگر $p = k$: عنصر x را برمی‌گردانیم.
 - اگر $p > k$: به طور بازگشتی عنصر k ام را در $A[1, 2, ..., p-1]$ پیدا می‌کنیم.
 - اگر $p < k$: به طور بازگشتی عنصر $k-p$ ام را در آرایه $A[p+1, ..., n]$ پیدا می‌کنیم.

عنصر میانه B (عنصر x) حداقل از $\frac{n}{5}$ عناصر کوچکتر و حداقل از $\frac{n}{5}$ عناصر بزرگتر است. بنابراین سه حالت قسمت ۵ در بدترین حالت از مرتبه $T(O(\frac{n}{5}))$ هستند. پس به کمک استقرا ثابت می‌کنیم که الگوریتم $f-select$ از مرتبه $O(n)$ می‌باشد.

