

۱) شایستگی آباری ۴۷۱۳۱۵
 طراحی الگوریتم

۱) ابتدا تمام خانه های آرایه را با یک صفر می بندیم تا خانه ها برابر ۰ را داشته
 شوند. برای این که Countingsort روی اعداد در بازه k جواب می دهد

Order آن $\Theta(n+k)$ است. که این جا $k=2$ است.
 For $t=0$ to $A.length-1$
 $A[t]++$

Counting Sort (A, B, 2)

For $t=0$ to $B.length-1$

$B[t]--$

حال می بینیم که B است. دو حلقه ای for n بار داریم و Counting Sort

که می بینیم $\Theta(n+k)$ است. ← این الگوریتم از مرتبه $\Theta(n)$ است

مثال
 $A : 1 \ 0 \ -1 \ -1 \ 0 \ 0 \ 0 \ 1 \ -1 \ -1$
 $B :$

$n = 10$
 $K = 2$
 App $\rightarrow A : 2 \ 1 \ 0 \ 0 \ 1 \ 1 \ 1 \ 2 \ 0 \ 0$
 $B :$

$C :$

	0	1	2
	←	←	←

↓

$C :$

	←	1	10
--	---	---	----

↓

$B :$

	1	2	3	4	5	6	7	8	9	10
	0	0	0	0	1	1	1	1	2	2

↓

$B-- :$

	-1	-1	-1	-1	0	0	0	0	1	1
--	----	----	----	----	---	---	---	---	---	---

۳) در زمان $O(n)$ حل می شود زیرا for های درون الگوریتم Counting sort

$K \leq n$ می رود مراحل کار به این ترتیب است که ابتدا اعداد آرایه را به بازه ی

۰ تا K که این جا $K=2$ است می بینیم سپس خانه های آرایه ی C را برابر ۰ می کنیم

سپس تعداد تکرار عنصر i در آرایه ی A را در خانه ی i از آرایه ی C ذخیره

می کنیم مثلاً اگر در A عدد ۳، ۴ بار آمده $C[3]$ برابر ۴ می شود

سیس از ابتدای آرایه شروع می‌کنیم و خانه‌های آن را با ۱-۱۱ ام جمع
می‌کنیم بدین شکل در خانه‌ی اول که ۰ است مقدار تکرار صفرها در خانه‌ی

بعدی مقدار تکرار صفرها یک‌ها در خانه‌ی بعدی مقدار ۰ هار یک‌ها در خانه‌ی بعدی
شکل آنتهای آرایه. حال از روی آرایه‌ی A، C، B را بر می‌کنیم

در حال حاضر هر عدد باید در خانه $C[i-1] + 1$ آرایه‌ی B

قرار گیرد. مثلاً عدد یک باید در خانه‌ی $C[0] + 1$ ام آرایه‌ی B $C[1]$ ام

آرایه‌ی B قرار گیرد پس به ازای هر عدد آرایه‌ی A جایش را از روی

صورتی $C[i]$ پیدا می‌کنیم و $C[i]$ را منهای یک می‌کنیم تا آن بعدی

در جای درست خود قرار گیرد. یعنی اگر عدد ۵ را در A بخواهیم در جای درست

خود بگذاریم $C[5] = 4$ باشد یعنی ۵ را در خانه‌ی ۴ ام B می‌گذاریم و

$C[5]$ را برابر ۳ می‌کنیم تا ۵ بعدی در خانه‌ی ۳ ام برود