

برای اعداد زیر به الگوریتم Insertion Sort پاسخ دهید:

۱۲، ۸ و ۱۱ - ۳ و ۹ - ۸ و ۱

پایه - در مرتب سازی درجی پایه آن است که بخشی از مساله
حل می شود و بعد زره زره این مجموعه بزرگ می شود.



نامرتب ۱۲، ۸ و ۱۱ - ۳ و ۹ - ۸ و ۱

اشاره گر i به اولین خانه آرایه اشاره می کند (نکته: یک عدد
مرتب هست و با خود یک صف مرتب را می سازد) حال
هر نامرتبی می کنیم مجموعه حل را بزرگتر کنیم.

۱۲، ۸ و ۱۱ - ۳ و ۹ - ۸ و ۱
↓
S

نامرتب آن یک مقدار ارتقا میابد ++ i ، (قرار است در
مجموعه حل خود S ، همان جدید ۸ - را درج insert
کنیم)

برای اینکار ۸ - را در key می بینیم ۸ - key
و صف ما یک جای خالی به خود می گنجد...

key = -1

۱۲، ۸ و ۱۱- و ۳ و ۹- و ۱

۵

هفتم در نام دوم هستیم عدد ۸- با آخرین عضو ۵ مقایسه
شود تا به تصمیم گرفته شود برای جای ایجاد شده کدام
یک از آن‌ها باید قرار گیرد. مقایسه $1 < 8-$ است
پس یک باید یک واحد شیفت داده شود به راست.
و حال ابتدای صف یک جای خالی ایجاد می شود و چون
اول صف است عدد ۸- بدون مقایسه ای درج می شود
یعنی:

shift

۱۲، ۸ و ۱۱- و ۳ و ۹- و ۱
۱۲، ۸ و ۱۱- و ۳ و ۹- و ۱ و ۸-
۸-

دوباره یک جای خالی

برای تصمیم گیری ایجاد شده

اقدام به درج صورت خواهد گرفت

چون به ابتدای صف رسیدیم

صف نامرتب

(۵) صف مرتب

۱۲، ۸ و ۱۱- و ۳ و ۹- و ۱ و ۸-

۱۲، ۸، ۱۱ - ۳ و ۹ - ۱ و ۸ -

گام بعدی ++A، چند بود؟ A = ابودوالان نوشتند.

۱۲، ۸، ۱۱ - ۳ و ۹ - ۱ و ۸ -

یعنی چه؟ یعنی در صف مرتب خود می خوانیم
 عنصر آخر را درج کنیم؛ پس چه می کنیم - گفتیم همانند
 مثال صف درسد که ناقص درست دانشجو را می گرفت و از
 صف خارج می کرد درست ۹ - را می گیریم و در Key
 می بینیم.

۹ ← Key
 ۱۲، ۸، ۱۱ - ۳ و ۹ - ۱ و ۸ -

در اینجا با خالی در آخر صف مرتب هستیم که می بینیم
 با مقایسه ۹ - با بلندترین صف مرتب یا عدد ۱ کدام بلندتر
 هست؟ ۱؛ پس عدد ۱ باید شیفست پیدا کند...

۱۲، ۸، ۱۱ - ۳ و ۹ - ۱ و ۸ -

حال ۹ - Key با ۸ - مقایسه می شود برای جای خالی
 ۸ - بلندتر است و باید شیفست پیدا کند...

12 و 8 و 11 - و 3 و 1 و 1 - و 9 -

در این گام از مرتب سازی در 9 - key به ابتدای صف
 اسید در جای مناسب قرار می گیرد. وقت کنید مجموعه ای حل یا 9
 ذره ذره دارد بزرگ می شود. و صف 9 یا صف آشفته
 ذره ذره ما دارد کوچک می شود.

برای عددها در گام بعد یعنی:

گام 2 3 - key مقدار 3 می شود

12 و 8 و 11 - و 3 و 1 و 1 - و 9 -

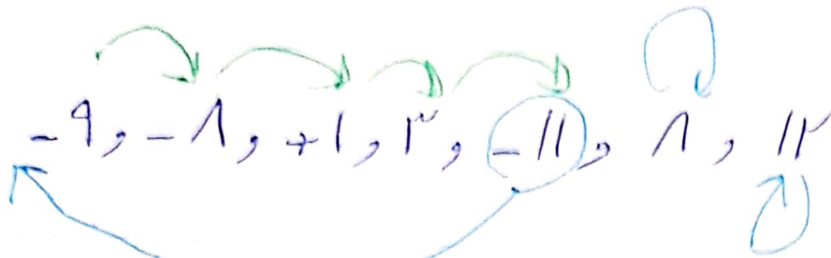
$key = +3$

حال چه؟ $+3$ از یک عدد آفرین عفو عنصر
 مرتب بزرگ تر است و در جای خود می ماند پس

12 و 8 و 11 - و 3 و 1 و 1 - و 9 -

می توانند گام بعد را حدس بزنند: 11 - پس از ریفته شدن
 در صف key به ابتدای صف پس می آید و تعداد صف
 ما چند تا خواهد بود؟ 4 تا $+3$ و 1 و 1 - و 9 - باید
 اواخر صف بیدارند یا به عبارتی حبابی باید به ابتدای
 صف در هر مرحله انتقال داده شود.

و پس از سه نام به هم می‌گیریم؛ در نهایت صفیبت به شکل زیر خواهد رسید:



نهایت: 12, 8, 3, 1, -8, -9, -11

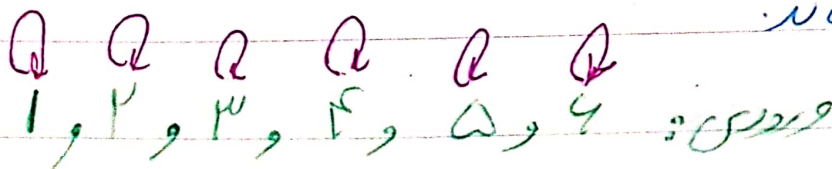
و جنب آنچه مهم نیست برای ما در این نام چه نیست؟

الف - مفهوم *Inplace*

آرایه جدید عنصر داشته و نیازی به طول K عنصر باشد
 آرایه هر طولی داشته باشد $1 \leq n$ و تنها برای جابه
 جا کردن اعداد به یک key نیاز داریم و همان آرایه
 ورودی کافی است... به الگوریتم‌هایی که نسبت
 به فضای ورودی مساله فضای گفلی ثابتی داشته
 باشند (مسئله از سبب ورودی مساله) درجای گویند.

ب - مرتبه زمانی

سه حالت بهترین حالت اگر یک صف مرتب
 داشته باشیم آرایه کار را با یک بار تپاسش از ابتدا
 به انتها و با جاگذاشتن هر عنصر در جای خود پایان
 خواهد رساند.



اینجا مفهوم بهتری بیان شده
 $key \leftarrow 3$
 ۱ ۲ ۳ ۴ ۵ ۶

لحظه‌ای که در این حالت صفر خواهد بود در بدترین حالت یعنی در تابع مرتب‌سازی درجی

```
def InsertionSort (A
  for i : 0 to n :
```

```
    key ← i
    j ← i - 1 // S[-1]
    while key < A[j]:
        shift A[j] → A[j+1]
        j--
```

← دست عدد توسط
 ناظم گرفته می‌شود
 ← شیف
 ← کمترین
 حالت شیف
 ندارد اما بدترین
 حالت خواهد بود *

```
    A[j] ← key
```

← عدد در جایگاه صحیح
 خود را می‌نویسد

در بدترین حالت n شیف داریم و n شیف یعنی
 کن آرایه معکوس مرتب شده باشد و با گرفتن دست
 نفر n ام باید آن را به اول صف ببریم در این حالت
 کران بالایی برای این الگوریتم از مرتبه $O(n^2)$ است