

بسم الله الرحمن الرحيم

موضوع پروژہ :

# Quick Sort

نام تهیه کننده :  
حمید امید

نام استاد مربوطه :  
جناب آقای سهامی

با تشکر و سپاس فراوان

## مرتب سازی سریع (sort Quick)

مرتب سازی سریع از راهبرد تقسیم و حل استفاده می کند . این راهبرد یک روش بازگشتی است که در آن مسئله ای که باید حل شود به مسئله کوچکتر تقسیم میگردد .

عنصری از آرایه به نام محور انتخاب می شود و سپس دنباله ای از تعویض ها صورت میگیرد تا عناصر کوچکتر از محور در سمت چپ آن و عناصر بزرگتر از محور در سمت راست آن قرار گیرند . اکنون عنصر محور در جای اصلی خود در آرایه قرار دارد . و آرایه به دو آرایه کوچکتر تقسیم شده که می توان آنها را با همین روش مرتب کرد .

بدترین حالت مرتب سازی سریع زمانی است که آرایه مرتب باشد یا به ترتیب معکوس مرتب شده باشد . زمان در بدترین حالت  $n^2$  یا  $O(n \log n)$  میانگین است .

8 2 13 5 14 3 7

3 2 7 5 8 14 13

---

8 2 13 5 14 3 7

3 2 7 5 8 14 13

2 3 7 5

---

8 2 13 5 14 3 7

3 2 7 5 8 14 13

2 3 7 5

---

8 2 13 5 14 3 7

3 2 7 5 8 14 13

2 3 7 5

5 7 -

---

8 2 13 5 14 3 7

3 2 7 5 8 14 13

2 3 7 5

5 7 -

8 2 13 5 14 3 7

3 2 7 5 8 14 13

2 3 7 5

5 7 -

---

8 2 13 5 14 3 7

3 2 7 5 8 14 13

2 3 7 5 13 14

5 7 -

---

8 2 13 5 14 3 7

3 2 7 5 8 14 13

2 3 7 5 13 14

5 7 -

---

2 3 5 7 8 13 14

++C پیاده سازی (مرتب سازی سریع)

```
Void quicksort(T a[],int first,int last)
```

```
{
```

```
Int pos;
```

```
if(first < last)
```

```
{
```

```

Split(a,first,last,&pos);

Quicksort(a,first,pos-1);

Quicksort(a,pos+1, last);

}

}

//*****

Template

Void split(T a[] , int first , int last , int *pos)

{

Int left = first , right =last , pivont = a[first];

T t;

While(left < right)

{ While(a,[right] >pivot)

Right--;

While(left

Left ++;

If(left < right)

{

T=a[left];

A[left]=a[right];

A[right]=t;

}

}

*pos =right;

A[first]=a[*pos];

A[*pos]=pivot; }

```

## پیاده سازی (مرتب سازی سریع) پاسکال

```
Procedure quicksort(var a:array ; right,left:integer);  
I , j , t , v: ineger ;  
Begin  
If right > left then  
Begin  
V:=a[right];  
I:=left-1;  
J:=right;  
Repeat  
Repeat  
I:=i+1;  
Until a[i] >=v;  
Repeat  
J:=j-1;  
Until a[j] <=v;  
T:=a[i];  
A[i]:=a[j];  
A[j]:=t;  
Until j < = I ;  
A[j] :=a[i];  
A[i] :=a[right];  
A[right]:=t;  
Quicksort(a,left,i-1);  
Quicksort(a,i+1,right);  
End;  
End;
```

بررسی مرتب سازی سریع (sort Quick)

$$W(n) = n(n-1)/2 \rightarrow w(n) \leq n(n-1)/2 = \theta(n^2)$$

برای پارتیشن  $T(n) = n-1$

$$A(n) = 1/n \sum A(p-1) + A(n-p) + n-1$$

$$= 1/n \sum A[(p-1) + A(n-p)] + n-1$$

$$A(0) + A(n-1) + A(1) + A(n-2) + \dots + A(n-2) + A(n-1) + A(1) + A(0)$$

$$= 2/n \sum A(p-1) + n-1$$

$$nA(n) = 2 \sum A(p-1) + n(n-1)$$

$$(n-1)A = 2 \sum A(p-1) + (n-1)(n-2)$$

$$nA(n) - (n-1)A(n-1) = 2A(n-1) + 2(n-1)$$

$$nA(n) - (n+1)A(n-1) = 2(n-1)$$

$$A(n)/n+1 = a(n-1)/n + 2(n-1)/n(n+1)$$

$$N = A(n) / n+1 \rightarrow An = an-1 + 2(n-1) / n(n+1)$$

$$n-1 \sim n+1$$

$$\sim an = an-1 + 2/n$$

$$= an-2 + 2/n-1 + 2/n$$

$$= an-3 + 2/n-2 + 2/n-1 + 2/n$$

$$\rightarrow an \sim 2 \lg n$$

$$A(n) / n+1 = 2 \lg n \rightarrow A(n) = 2(n+1) \lg n$$

$$= 2(n+2) \lg n * \ln 2$$

$$= 1.38(n+1) \lg n$$

$$= \theta(n \lg n)$$