



دانشکده مهندسی کامپیوتر
جزوه درس
ساختمان‌های داده

استاد درس: سید صالح اعتمادی

پاییز ۱۳۹۸

جلسه ۵

تقسیم و حل

فاطمه احمدی - ۱۳۹۸/۷/۱۳

یکی از مسائلی که در آن از الگوریتم تقسیم و حل استفاده می شود مسئله پیدا کردن مینیمم است که به صورت زیر انجام می شود:

Data: Ages of Students

Result: how to write algorithm to find youngest student

initialization;

Min(nums,low,high)

if $low = high$ **then**

| return nums(high);

end

mid <- low + (high - low)/2;

return Math.Min(Math.Min(nums,low,mid),Math.Min(nums,mid,high));

Algorithm 1: How to find Min with divide and conquer

پیشگیری زمانی الگوریتم فوق از رابطه زیر پیروی می کند:

$$T(n) = 2T(n/2) + c$$

توجه کنید که الگوریتم فوق الگوریتم از نوع تقسیم و حل است و مسئله پیدا کردن مینیمم دارای جوابی با الگوریتم حریمانه نیز هست.

یکی دیگر از مسائلی که با الگوریتم تقسیم و حل قابل حل است مسئله تعیین بودن یک آیتم در یک گروه

از آیتم ها است که در زیر بررسی شده است:

```

Data: Array of Numbers
Result: v is in Array or not ?
initialization;
Find(v,nums,low,high)
if high < low then
  | return false;
end
mid <- low + (high - low)/2;
if nums(mid) = v then
  | return true;
end
return (Find(v,nums,low,mid - 1) or Find(v,nums,mid + 1,high))

```

Algorithm 2: Checking v is in the Array or not

پیچیدگی زمانی الگوریتم فوق نیز از رابطه زیر پیروی می کند:

$$T(n) = 2T(n/2) + c$$

الگوریتم تقسیم و حل بیشترین کاربرد را در آرایه های مرتب شده دارد در این حالت الگوریتم در پیچیدگی زمانی $\log(n)$ انجام می شود.

در شبه کد زیر مسئله قبل را با آرایه مرتب شده بررسی میکنیم:

```

Data: SortedArray of Numbers
Result: v is in Array or not ?
initialization;
FindSortedArray(v,nums,low,high)
if high < low then
  | return false;
end
mid <- low + (high - low)/2;
if nums(mid) = v then
  | return true;
end
if v < nums(mid) then
  | return FindSortedArray(v,nums,low,mid);
else
  | return FindSortedArray(v,nums,mid + 1,high);
end

```

Algorithm 3: Checking v is in the SortedArray or not

پیچیدگی زمانی الگوریتم فوق نیز از رابطه زیر پیروی می کند:

$$T(n) = T(n/2) + c$$

مسئله ضرب دو چند جمله ای نیز با الگوریتم تقسیم و حل قابل بررسی است و این روش نیز خود به دو

حالت ساده و سریع انجام می شود که با استفاده از روش سریع پیچیدگی محاسباتی کم می شود به طوری که به جای ۴ عملیات ضرب با ۳ عملیات ضرب به جواب می رسیم و رابطه اول زیر به رابطه دوم تبدیل می شود این در حالی است که پیچیدگی در حالت عادی در زمان n^2 انجام می شود:

$$T(n) = 4T(n/2) + kn$$

$$T(n) = 3T(n/2) + kn$$

روش اول ضرب را به صورت زیر انجام می دهد:

$$A(x) = a_1x + a_0$$

$$B(x) = b_1x + b_0$$

$$C(x) = a_1b_1x^2 + (a_1b_0 + a_0b_1)x + a_0b_0$$

روش سریع نیز به صورت زیر انجام می شود:

$$C(x) = a_1b_1x^2 + ((a_1 + a_0)(b_1 + b_0) - a_1b_1 - a_0b_0)x + a_0b_0$$

Bibliography