



ساختمان داده‌ها و الگوریتم‌ها

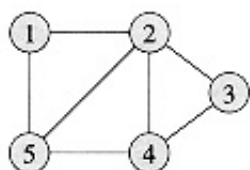
نیم‌سال اول ۹۹-۰۰
مدرس: مسعود صدیقین

یادآوری جلسه بیستم

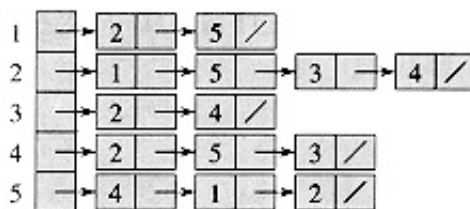
گراف

در جلسه قبلی، مقدماتی راجع به نظریه گراف و همچنین روش‌های پیاده‌سازی گراف معرفی کردیم. برای پیاده‌سازی گراف، سه روش مختلف معرفی کردیم که دو تا از آنها عبارت بودند از:

- ماتریس مجاورت: یک ماتریس A با ابعاد $n \times n$ که در آن، $A[i][j]$ برابر با یک است، اگر و تنها اگر یالی بین راس‌های i و j در گراف وجود داشته باشد و در غیر این صورت صفر است.
- لیست مجاورت: یک آرایه با اندازه n که اندیس i ام آن یک اشاره‌گر به لیستی شامل همسایه‌های راس i ام است.



(a)



(b)

| | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | 0 | 1 | 0 | 0 | 1 |
| 2 | 1 | 0 | 1 | 1 | 1 |
| 3 | 0 | 1 | 0 | 1 | 0 |
| 4 | 0 | 1 | 1 | 0 | 1 |
| 5 | 1 | 1 | 0 | 1 | 0 |

(c)

در جلسه امروز، راجع به پیمایش عمق نخست گراف صحبت خواهیم کرد. جستجوی عمق نخست به صورت زیر است:

```
DFS(v_i){
    mark[v_i] = true;
    for every v_j ∈ N(v_i)
        if (!mark[v_j])
            DFS(v_j)
}
```

در این قطعه کد، $mark$ یک آرایه با اندازه n است که در ابتدا مقدار همه خانه‌های آن $false$ است.

تمرین: برای گراف زیر، یک بار DFS را از راس ۰ اجرا کنید و مشخص کنید آخرین راسی که DFS آن صدا زده می‌شود کدام است (فرض کنید در حلقه for همسایه‌ها به ترتیب اندیس از کمتر به بیشتر بررسی می‌شوند). همچنین مشخص کنید زمان اجرای DFS برای یک گراف با n راس و m یال، در حالت‌هایی که گراف با ماتریس مجاورت و لیست مجاورت پیاده شده باشد چقدر است. پاسخ‌های خود را به این آدرس ارسال کنید.

