



## ساختمان داده‌ها و الگوریتم‌ها

نیم‌سال اول ۱۴۰۰-۱۴۰۱  
مدرس: مسعود صدیقین

رویا دانشی

گراف

یادآوری جلسه بیست‌ویکم

در جلسه قبل مقدماتی راجع به نظریه گراف و روش‌های پیاده‌سازی گراف بیان شد. ابتدا به تعریف مفاهیم مربوط به گراف می‌پردازیم:  
**گراف:** هر گراف به‌صورت یک دوتایی  $\langle V, E \rangle$  تعریف می‌شود که  $V$  مجموعه راس‌های گراف و  $E$  مجموعه یال‌های گراف است.

هر یال گراف به‌صورت دوتایی از راس‌ها  $(V_j, V_i)$  می‌باشد.

**همسایه:** راس  $V_i$  و  $V_j$  همسایه هستند اگر بین آن‌ها یال باشد.

**درجه:** درجه راس  $V_i$  معادل تعداد همسایه‌های آن است و آن را با  $d(V_i)$  نشان می‌دهند.

**گشت:** دنباله‌ای از راس‌های گراف به‌صورت  $V_{l_1}, V_{l_2}, \dots, V_{l_k}$  به‌طوری‌که به ازای هر  $i$ ،  $(V_{l_i}, V_{l_{i+1}})$  عضو یال‌های گراف باشد.

**گذر:** گشت بدون یال تکراری است.

**مسیر:** گذر بدون راس تکراری است.

**دور:** مسیری که ابتدا و انتهای یکسانی داشته باشد. اندازه یک دور برابر تعداد یال‌های آن دور است.

**طوقه:** یالی که ۲ سر آن راس یکسان باشد.

**یال موازی:** دو یال موازی هستند، اگر سرهای یکسان داشته باشند.

**گراف همبند:** گرافی که بین هر دو راس آن یک مسیر وجود دارد.

**مولفه همبندی:** زیرگراف بیشینه و همبند از گراف اصلی می‌باشد.

**راس برشی:** راسی است که حذف آن تعداد مولفه‌های همبندی را افزایش می‌دهد.

**یال برشی:** یالی که حذف آن تعداد مولفه‌های همبندی را افزایش می‌دهد.

**گراف مکمل:** گرافی که مجموعه راس‌های آن، مجموعه راس‌های گراف اصلی بوده و شامل یال‌هایی است که در گراف اصلی وجود ندارند. اگر دو گراف  $G = \langle V, E \rangle$  و  $G' = \langle V, E' \rangle$  مکمل یکدیگر باشند، رابطه  $|E| + |E'| = \binom{n}{2}$  بین تعداد یال‌های آن‌ها برقرار است.

**گراف کامل:** گرافی که بین هر دو راس آن یک یال وجود دارد.

**زیرگراف:** گرافی که مجموعه راس‌های آن زیرمجموعه‌ای از راس‌های گراف اصلی و مجموعه یال‌های آن زیرمجموعه‌ای از یال‌هایی است که بین راس‌های موردنظر در گراف اصلی وجود دارد.

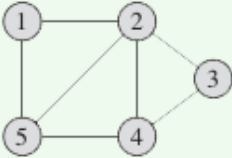
**زیرگراف القایی:** زیرگرافی که شامل زیرمجموعه‌ای از راس‌های گراف اصلی و تمام یال‌های بین آن‌ها است.

**درخت:** گراف همبند بدون دور را درخت می‌گوییم.

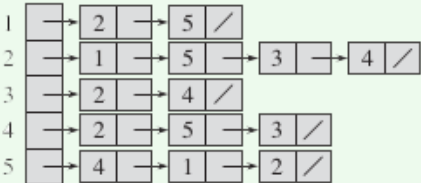
حال به بررسی دو روش پیاده‌سازی گراف می‌پردازیم:

- **ماتریس مجاورت:** برای ذخیره‌سازی گراف از یک ماتریس با ابعاد  $n \times n$  استفاده می‌کنیم. بنابراین حافظه مورد نیاز از مرتبه  $\theta(n^2)$  می‌باشد. در این روش پیاده‌سازی بین دو راس  $i$  و  $j$  یال وجود دارد اگر و تنها اگر  $A[i, j] = 1$ . بدین ترتیب می‌توان در زمان  $O(1)$  بررسی کرد که آیا بین ۲ راس  $i$  و  $j$  یال وجود دارد یا خیر. همچنین در زمان  $O(n)$  می‌توان لیست همسایه‌های یک راس را به دست آورد.

	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



- **لیست مجاورت:** برای ذخیره‌سازی گراف، یک آرایه به اندازه  $n$  در نظر می‌گیریم که اندیس  $i$ ام آن یک اشاره‌گر به لیستی شامل همسایه‌های راس  $i$ ام است. حافظه مصرفی این روش از مرتبه  $O(n + 2m)$  می‌باشد که  $m$  تعداد یال‌ها و  $n$  تعداد راس‌ها می‌باشد. به کمک این روش در زمان  $O(d_i)$  می‌توان بررسی کرد که بین دو راس  $i$  و  $j$  یالی وجود دارد یا خیر. به دست آوردن لیست همسایه‌های راس  $i$ ام نیز از مرتبه  $O(d_i)$  می‌باشد.



**پرسش:** ماتریس مجاورت و لیست مجاورت مربوط به گراف شکل زیر را به دست آورید. پاسخ‌های

خود را به **این لینک** ارسال کنید.

