



تمرین شماره 5

Graph



ساختمان های داده و الگوریتم - پاییز 1401

مهلت تحویل:

دانشکده مهندسی برق و کامپیوتر

طراح تمرین: **علیرضا آقایی**

1401/10/20، ساعت: 23:59

استاد: دکتر هشام فیلی

### سوال ۳ ( ۱۵ نمره)

بهترین استراتژی برای دوقل این است که در ابتدا تعدادی مرحله به رئوس بالاتر برود. سپس به پایین ترین راس موجود

در زیردرخت راس کنونی خود رفته و در آنجا تا زمان رسیدن به قل صبر کند.

با استفاده از dfs، پایین ترین راس (راسی با بیشترین عمق در زیر درخت هر راس)، فاصله ی رأس ها را از راس ریشه و

راسی که دوقل روی آن قرار دارد را به دست می آوریم. همچنین با توجه به ریشه دار بودن درخت از راس ۱، پدر هر

راس را نیز با استفاده از dfs مشخص می کنیم.

حال برای به دست آوردن بهترین جواب، روی پدرهای راسی که دوقل روی آن قرار دارد حرکت می کنیم، و به ازای

هر کدام بررسی می کنیم که آیا فاصله ی دوقل از آن راس کمتر از فاصله ی به قل تا آن راس هست یا خیر. اگر بود،

آن گاه مقدار مجموع فاصله ی به قل تا آن راس و ارتفاع پایین ترین راس زیر درخت آن راس را با جوابی تا به حال

داشتیم، ماکزیمم می گیریم. در نهایت به جواب مورد نظر برای سوال می رسیم.

#### سوال ۴ ( ۱۵ نمره)

از دو راس  $a$  و  $b$ ، یک بار الگوریتم  $bfs$  را اجرا می‌کنیم. سپس به ازای هر دو راسی که به یک دیگر یال ندارند، مانند  $s$  و  $c$ ، مقدار فاصله آن‌ها از  $a$  و  $b$  رو با هم جمع کرده و سپس به علاوه‌ی ۱ ( یال جدید) می‌کنیم، اگر این مقدار از فاصله راس  $a$  از  $b$  کمتر نشد، به این معنی است که یال جدید یک یال مناسب برای اضافه شدن به گراف است. برای بررسی همه‌ی یال‌ها نیز هر جفت راس را باید بررسی کرد که بنابراین پیچیدگی زمانی الگوریتم برابر با  $O(n^2)$  می‌شود.

#### سوال ۵ ( ۱۵ نمره)

ابتدا طبق الگوریتم ۱، رئوس متعلق به دور را در گراف پیدا می‌کنیم. جواب مساله برای این رئوس برابر با صفر است. سپس تمام راس‌های مربوط به این دور را تبدیل به یک راس کرده، و روی گراف جدید یک  $dfs$  با شروع از این راس جدید اجرا می‌کنیم. با این کار فاصله‌ی سایر رئوس از این دور از گراف نیز به دست می‌آید. پیچیدگی زمانی این الگوریتم با توجه به این که تعداد یال‌ها برابر با  $n$  است برابر با  $O(n)$  می‌شود.

الگوریتم ۱: روی گراف یک  $DFS$  با کمی تغییر اجرا می‌کنیم. این موضوع را در نظر می‌گیریم که یک  $back-edge$  قرار است بینیم و آن را بین دو راس که در حال اجرای  $DFS$  راس پایینی هستیم اما  $DFS$  راس بالایی هم تمام نشده است و همچنین این راس بالایی،  $parent$  آن نمی‌باشد می‌بینیم. بنابراین اگر در پیمایش یال‌ها در  $DFS$  به یالی برسیم که راس دیگرش  $mark$  شده است اما پدر راس فعلی نیست، راس دیگر دور را پیدا کرده‌ایم، و رئوسی که بین این راس تا راس بالایی در مسیر رسیدن به ریشه هستند، اعضای دور محسوب می‌شوند. با دنبال کردن پدرها می‌توانیم آنها را جدا کنیم.