

# پنج‌شنبه‌ی پانزدهم

اثبات کنید در جهان بهترین هستید...

شاید برخی گمان کنند دو پنج‌شنبه‌ی اخیر بسیار آرام برگزار شده‌اند و علاقمندان پنج‌شنبه‌های سخت شاهد شکوهی که از چنین رقابت بزرگی انتظار داشته‌اند، نبوده‌اند. وظیفه‌ی خود می‌دانیم (سردبیر و سایر اعضای گروه تحریریه‌ی پنج‌شنبه‌های سخت) تا خوانندگان گرامی را از تلاش‌های بی‌نظیر دو هفته‌ی اخیر برای پاسخ به پنج‌شنبه‌ی چهاردهم آگاه کنیم. اعضای تیم فنی و شرکت‌کنندگان روزها و شب‌ها به دنبال یافتن بهترین جواب تلاش کرده‌اند. منابع بی‌شماری توسط دال زیر و رو شده‌اند تا ایده‌هایی برای بهترین جواب این مسئله پیدا شوند. با این وجود، همچنان جواب راضی‌کننده‌ای برای پنج‌شنبه‌ی چهاردهم ارائه نشده است و تلاش برای یافتن بهترین جواب این مسئله همچنان ادامه دارد. برخی از یادداشتهای سردبیر و دال را در مورد پنج‌شنبه‌ی چهاردهم برای خوانندگان گرامی در ادامه تکرار می‌کنیم.

## در مورد پنج‌شنبه‌ی چهاردهم — قسمت پنجم

هنوز در پنج‌شنبه‌ی چهاردهم به نتیجه‌ی تئوری خوبی نرسیده‌ام. اما به نظرم جای خالی برای ارائه‌ی یک الگوریتم عملی خوب وجود دارد. ضریب بهترین الگوریتم تقریبی که تا این لحظه دیده‌ام  $\sqrt{m}$  است (فرض کنید  $m$  اندازه‌ی بهترین جواب باشد) و الگوریتم تصادفی خوبی برای این مسئله ندیده‌ام. بنابراین، الگوریتم ترکیباتی دقیق یا غیر دقیق خوبی برای این مسئله احتمالا وجود ندارد. از این رو، پیشنهاد می‌کنم مسیر زیر را برای این مسئله در پیش بگیریم تا یک الگوریتم غیر دقیق خوب برای این مسئله ارائه دهیم.

۱ اگر که تعداد خط‌ها کم باشد الگوریتم آقای یزدان‌پناه بد عمل نمی‌کند. الگوریتم دوست دال نیز جواب نسبتا خوبی برای ورودی‌های مختلف پیدا می‌کند. با استفاده از این دو الگوریتم (و شاید چند ایده‌ی دیگر) می‌توان حد پایینی برای تعداد بیشینه‌ی نقطه‌های غیر هم‌خط بدست آورد.

۲ در مقاله‌ای که قبلا به آن اشاره کردم [۱]، سه قاعده برای کاهش اندازه‌ی مسئله پیشنهاد می‌شوند. با توجه از حد بدست آمده در گام قبل، می‌توان از این سه قاعده برای کاهش اندازه‌ی مسئله سود جست.

۳ برای نقطه‌های باقی مانده باید جواب خوبی بدست آورد. یک راه استفاده‌ی دوباره از الگوریتم‌های گام ۱ است. راه دوم، در نظر گرفتن تعداد نقطه‌های حذف شده به جای نقطه‌های انتخاب شده است (مانند الگوریتم آقای یزدان‌پناه). در صورتی که هر سه نقطه‌ای که با هم خط هستند را در نظر بگیرید، حتما حداقل یکی از آنها باید حذف شود. در واقع تعدادی زیر مجموعه‌ی سه تایی دارید که باید به شکلی نقطه‌ها را انتخاب کنید که از هر یک از این مجموعه‌ها حداقل یک نقطه انتخاب شود. این مسئله، در حالت کلی «Hitting Set» نامیده می‌شود. برای این مسئله، می‌توان یک الگوریتم ساده با ضریب تقریب سه ارائه داد؛ یعنی حداکثر سه برابر بیش از تعداد کمینه رأس حذف کرد.

اما برای اینکه عملکرد الگوریتم خود را نشان دهیم، احتیاج به داده‌های ورودی متنوع‌تری نیاز داریم.

## در مورد پنج‌شنبه‌ی چهاردهم — قسمت چهارم

پیشنهادی از طرف دال به آقای یزدان‌پناه: در الگوریتم شما اگر شیوه‌ی انتخاب رأسی که قرار است حذف کنید را عوض کنید، می‌توان اثبات کرد که ضریب تقریب الگوریتم شما  $\log n$  است. البته، شاید این تغییر امتیاز برنامه‌ی شما را بهبود ندهد. تغییر این است که به جای اینکه نقطه‌ای را حذف کنید که در کمترین خط ظاهر شود، نقطه‌ای را حذف کنید که زوج‌های هم‌خط آن کمینه باشد. زوج‌های هم‌خط یک نقطه، تعداد همه‌ی زوج نقطه‌هایی از نقطه‌های باقی مانده است که با آن نقطه در یک خط قرار دارند. برای نمونه، اگر یک نقطه با چهار نقطه‌ی دیگر هم‌خط باشد، تعداد زوج‌های هم‌خط آن نقطه در آن خط برابر شش است.

## در مورد پنج‌شنبه‌ی چهاردهم — قسمت سوم

راه حل حریصانه‌ی این مسئله، ساختن یک زیر مجموعه‌ی  $S$  از نقطه‌های ورودی  $P$  است: هر نقطه از  $P$  را به  $S$  اضافه می‌کنیم که با هیچ دو نقطه‌ی دیگری از  $S$  هم خط نباشد. این جواب اگر چه به نظر می‌رسد در واقعیت خوب عمل کند، دارای ضریب تقریب  $O(\sqrt{m})$  است [۲] (اندازه‌ی بهترین جواب است). می‌توان مجموعه‌ای مثال زد که اندازه‌ی  $S$  به دست آمده از این الگوریتم ریشه‌ی درجه‌ی دوم اندازه‌ی جواب بهینه باشد. الگوریتم آقای یزدان‌پناه به نظر می‌رسد بهتر از الگوریتم اول عمل کند، اما باید در مورد ضریب تقریب آن بررسی بیشتری انجام داد که آیا می‌توان گفت همواره مجموعه‌ی بدست آمده، ضریبی از جواب بهینه است یا خیر.

اکنون در حال مطالعه‌ی مسئله‌ی Hitting Set هستیم و شاید بتوان با استفاده از آن مسئله جواب بهتری برای پنج‌شنبه‌ی چهاردهم یافت. لازم به اشاره است که مسئله‌ی پنج‌شنبه‌ی چهاردهم به مسئله‌ی Hitting Set قابل تبدیل است [۱].

1. V. Froese, I. Kanj, A. Nichterlein, R. Niedermeier, "Finding Points in General Position," pp. 7–14 in *The Canadian Conference on Computational Geometry* (2016).
2. C. Cao, *Study on Two Optimization Problems: Line Cover and Maximum Genus Embedding*, MSc Thesis, Texas A&M University (2012).

## در مورد پنج‌شنبه‌ی چهاردهم — قسمت دوم

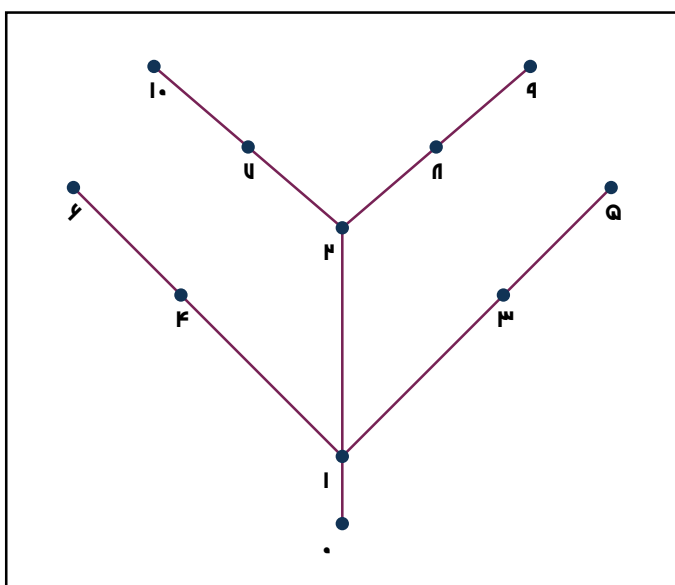
نکته‌ای که در مورد الگوی ورودی و خروجی در پنج‌شنبه‌ی یازدهم و چهاردهم وجود دارد این است که خروجی پنج‌شنبه‌ی یازدهم مستقیماً می‌تواند به پنج‌شنبه‌ی چهاردهم داده شود. برای مثال، اگر  $s_{11}$  جواب پنج‌شنبه‌ی یازدهم باشد و  $s_{14}$  جواب برنامه‌ی چهاردهم باشد و 00 یک نمونه‌ی ورودی پنج‌شنبه‌ی یازدهم باشد، می‌توان با استفاده از دستور زیر، زیرمجموعه‌ای از نقطه‌های غیر هم‌خط را برای نمونه‌ی ورودی یافت:

```
$ ./s11 <00 | ./s14
```

اما در مورد نمونه‌هایی که در پنج‌شنبه‌ی چهاردهم در نظر گرفته شده‌اند باید توجه شما را به مسئله‌ی دیگری جلب کنم، مسئله‌ی «No-Three-in-Line». در این مسئله یک شبکه‌ی  $n \times n$  در نظر گرفته می‌شود و هدف آن است که تعدادی نقطه انتخاب شوند که هیچ سه نقطه‌ای از آنها هم‌خط نباشند. این مسئله حالت خاصی از پنج‌شنبه‌ی چهاردهم (یا مسئله‌ی «General Position Subset Selection») است. در مورد مسئله‌ی «No-Three-in-Line»، الگوریتم قطعی سریعی وجود ندارد ولی اندازه‌ی بهترین جواب آن تا  $n = 47$  محاسبه شده است؛ برای این اندازه‌ها، بهترین جواب  $2n$  است. یعنی باید  $2n - 2$  نقطه حذف کرد تا هیچ سه نقطه‌ای در یک خط قرار نداشته باشند. برای توضیحات بیشتر به صفحه‌ی <http://oeis.org/A272651> و پیوندهای آن مراجعه کنید.

## در مورد پنج‌شنبه‌ی چهاردهم — قسمت اول

شکل زیر را یک خوشه می‌نامیم. به راحتی می‌توان نشان داد در یک خوشه، ۹ نقطه‌ی غیر هم‌خط وجود دارند. سه کپی از یک خوشه در نظر بگیرید و آنها را از نقطه‌ی صفرشان به هم بچسبانید به طوری که هیچ سه نقطه‌ای بین خوشه‌های مختلف هم‌خط نباشند (با تعیین زاویه‌ی مناسب برای چسباندن خوشه‌ها می‌توان این کار را کرد). اگر نقطه‌های حاصل به الگوریتمی که آقای یزدان‌پناه ارائه دادند به عنوان ورودی داده شوند، در گام اول نقطه‌ی صفر مشترک سه خوشه حذف می‌گردد. اما با حذف این نقطه، تعداد نقطه‌های غیر هم‌خط از حالت بهینه (که نقطه‌ی صفر در آن وجود دارد) یکی کمتر خواهد بود.



البته، حدس می‌زنم این الگوریتم در عمل بسیار خوب عمل کند اما بسیار جالب خواهد بود که به صورت نظری نشان داد این الگوریتم چه درصدی از نقطه‌ها را گزارش می‌دهد. همچنین، شاید در نظر گرفتن این حقیقت که اشتراک هر دو خط حداکثر یک نقطه هست نیز در دستیابی به الگوریتم بهتر کمک کند.