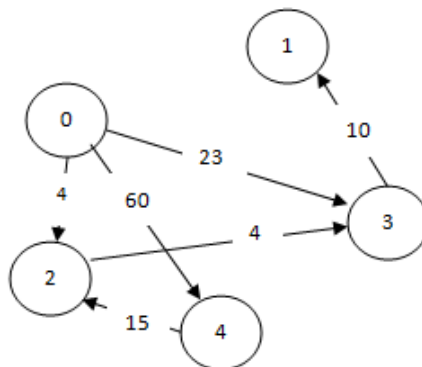


نکاتی در مورد انتخاب پروژه

- یک ایده حداکثر می تواند توسط 2 تیم یک نفره (یا دو نفره) انتخاب شود.
- پروژه ها باید تا شروع خردادماه انتخاب شوند.
- به پروژه ها حداکثر 3 نمره (از نمره کل) تعلق خواهد گرفت.
- تاریخ احتمالی تحویل پروژه ها متعاقبا اعلام می شود.
- برای دانستن جزئیات بیشتر در مورد پروژه ها، با ایمیل و یا مراجعه حضوری کسب اطلاع کنید.
- انتخاب پروژه ای غیر از پروژه های پیشنهادی در لیست زیر امکان پذیر است اما موضوع آن بایستی قبل از انتخاب به تایید برسد.
- استفاده از شی گرای object oriented programming در پیاده سازی پروژه ها الزامی است.
- در پروژه هایی که برای مسائل گراف تعریف شده اند، گراف ورودی بصورت فایل متنی بطوری که هر خط اطلاعات مربوط به یک یال را نشان می دهد به برنامه داده می شود. برای مثال در هر خط، ابتدا شماره راس مبدا، سپس شماره راس مقصد و سپس وزن یال آورده شده است. در اولین خط مشخص می شود که گراف جهت دارد است یا نه. در خط دوم تعداد رئوس گراف ذکر شده است. خطوط بعدی اطلاعات مربوط به یالها هستند.

```
0 # 1 for undirected, 0 for directed
5 # number of vertices
0 2 4 # edge from 0 to 2, with weight 4
0 4 60
0 3 23
2 3 4
3 1 10
4 2 15
```



لیست پروژه‌ها:

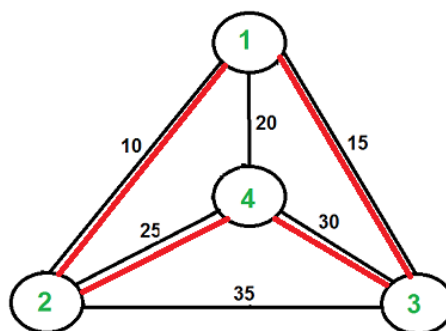
۱. شبیه‌سازی حرکت تصادفی در یک گراف (ولگشت). گراف بدون جهت $G = (V, E)$ داده شده است. ذره‌ی p در یکی از راسهای گراف $v \in V$ واقع شده است. این ذره در گراف شروع به قدم تصادفی می‌کند. در هر راس p به احتمال مساوی یالی را انتخاب می‌کند و حرکت خود را در آن یال ادامه می‌دهد. فرض کنید G همبند باشد می‌خواهیم بدانیم

- پس از چند حرکت کل راسهای گراف ملاقات می‌شود؟ برنامه‌نویس باید این آزمایش را برای گراف G و همه راسهای آن به کرات آزمایش کند و میانگین نتایج را برای هر راس چاپ کند. می‌خواهیم بدانیم آیا نقطه شروع تغییری در نتایج ایجاد می‌کند؟

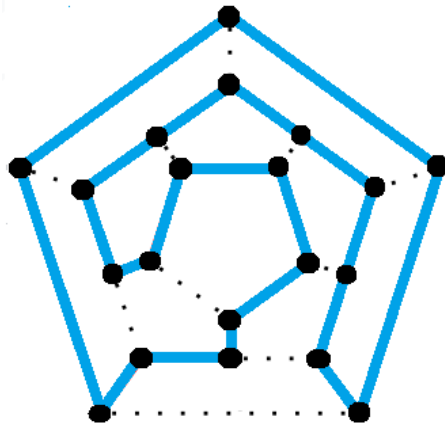
۲. شبیه‌سازی ترافیک در شبکه راه شهری. در این پروژه در واقع مشابه پروژه حرکت تصادفی در گراف است با این تفاوت که در این اینجا به جای یک ذره چندین ذره در گراف حرکت تصادفی انجام می‌دهند. ذره‌ها بطور تصادفی در یک نقطه از گراف ایجاد می‌شوند و شروع به حرکت تصادفی می‌کنند. هر ذره بعد از یک مدت (از قبل تعیین شده برای آن) به حرکت خود خاتمه می‌دهد و ناپدید می‌شود. می‌خواهیم بدانیم در گراف G

- کدام یالش سنگین‌ترین ترافیک را داراست؟ میزان ترافیک یالها با پارامترهای تصادفی مختلف سنجیده شود (تعداد ذرات، مدت حرکت، نقطه شروع).
- آزمون داده شده برای گراف داده شده شبیه‌سازی شود و نتایج ارائه شود.

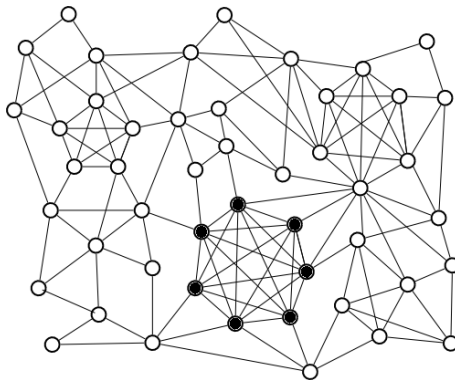
۳. مسئله فروشنده دوره‌گرد. گراف بدون جهت $G = (V, E)$ داده شده است. هر یال $e \in E$ یک وزن مثبت دارد. فرض کنید گراف داده شده مسیره‌های مختلف بین محله‌های یک شهر را نشان می‌دهد و وزنهای طول مسیره‌ها یا هزینه حمل و نقل در آن مسیره‌ها را نشان می‌دهد. یک فروشنده دوره‌گرد می‌خواهد از خانه خود شروع کند (راسی در گراف) و کل محله‌های شهر را بگردد و در نهایت به خانه خود باز گردد. چون هزینه مسیره‌ها متفاوت است، فروشنده دوره‌گرد می‌خواهد دوری را انتخاب کند که کمترین هزینه را برایش داشته باشد. در این پروژه هدف طراحی یک الگوریتم کارا برای پیدا کردن کم هزینه‌ترین دور با خواص گفته شده در گراف است. برنامه در نهایت باید یک دور پیشنهادی را با هزینه‌اش چاپ کند. در شکل زیر یک راه حل را برای شروع از راس شماره ۱ نشان می‌دهد.



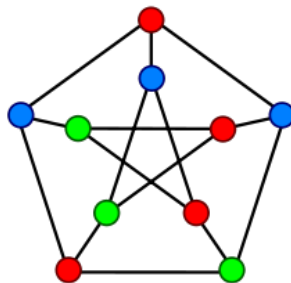
۴. داشتن دور همیلتونی گراف بدون جهت $G = (V, E)$ داده شده است. می‌خواهیم بدانیم آیا G دوری ساده دارد که شامل همه راسهای گراف باشد. به این گونه دورها، دور همیلتونی گفته می‌شود. شکل زیر یک گراف با دور همیلتونی مربوطه را نشان می‌دهد.



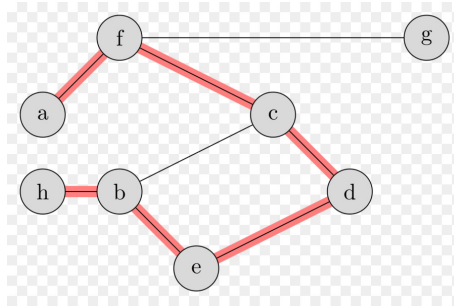
۵. پیدا کردن بزرگترین گروهک در گراف $G = (V, E)$ بدون جهت داده شده است. یک گروهک در G زیرمجموعه‌ای از رئوس است $S \subseteq V$ بطوریکه بین هر دو راس در S یک یال وجود دارد. می‌خواهیم بزرگترین گروهک در G را پیدا کنیم. شکل زیر یک گروهک را در گراف داده شده نشان می‌دهد.



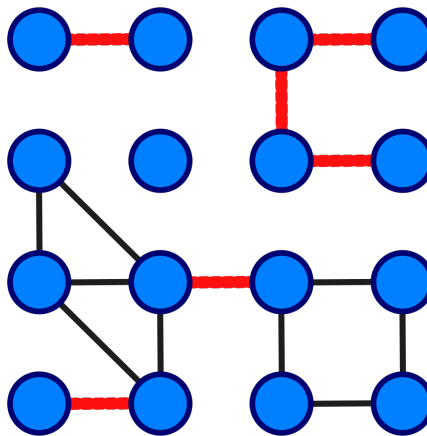
۶. رنگ آمیزی گراف $G = (V, E)$ بدون جهت داده شده است. می‌خواهیم رئوس گراف G را طوری رنگ کنیم که هیچ دو راس همسایه هم‌رنگ نباشند. بالاخص دنبال یک رنگ آمیزی با کمترین تعداد رنگ هستیم. شکل زیر یک رنگ آمیزی از گراف داده شده را نشان می‌دهد.



۷. پیدا کردن مسیری با طول k گراف بدون جهت $G = (V, E)$ داده شده است. می‌خواهیم بدانیم آیا در گراف G مسیری ساده وجود دارد که طولش k باشد؟ یعنی مسیر k یال داشته باشد و هیچ راسی در مسیر تکرار نشود. شکل زیر یک مسیر ساده با طول ۶ را در گراف داده شده نشان می‌دهد.



۸. پیدا کردن پل در گراف در گراف ساده $G = (V, E)$ به یال $e \in E$ پل گفته می‌شود هر گاه حذف e از گراف باعث شود که به تعداد مولفه‌های همبند G اضافه شود. به عبارت دیگر حذف e باعث شود که دو راس از G اتصالشان را از دست بدهند. در شکل زیر یالهایی که با رنگ قرمز مشخص شده اند، پل هستند.



الگوریتمی را پیاده‌سازی کنید که اطلاعات یک گراف را فایلی خوانده و همه پلهای گراف را پیدا کند.

۹. تحلیل گرافهای تصادفی و پدیده آستانه. گراف تصادفی G_p گرافی است که هر یال آن با احتمال $p \in [0, 1]$ ایجاد شده است. در این پروژه می‌خواهیم بدانیم برای گراف تصادفی G_p با تعداد رئوس زیاد، برای چه انتخابی از p گراف در آستانه داشتن یک ویژگی خاص است. در آستانه داشتن یک ویژگی بدین معنی است که اگر p را مقداری بسیار کم کاهش دهیم گراف ویژگی مربوطه را به احتمال زیاد نخواهد داشت ولی برای p گراف به احتمال زیاد ویژگی داده شده را دارد. چند نمونه از ویژگیهای مورد نظر در لیست زیر داده شده است.

- داشتن n مثلث
- همبند بودن

۱۰. نسبت تلفنی. برنامه لیستی از تماسهای تلفنی مختلف را از فایلی می‌خواند. لیست حاوی یک سری تماس تلفنی است که بصورت زوج شماره تلفن داده شده است.

23451134 - 34532212
 49822231 - 12334245
 98797088 - 23874098
 ...

شماره تلفن x و y نسبت درجه یک دارند اگر تماس تلفنی $x - y$ یا $y - x$ در لیست تماسها وجود دارد. شماره تلفن x و y نسبت درجه دوم دارند اگر نسبت درجه اول نداشته باشند و شماره تلفن z وجود داشته باشد که نسبت درجه اول با هر دوی x و y داشته باشد. به همین ترتیب شماره تلفن x و y نسبت درجه i ام دارند اگر نسبت درجه کمتر از i نداشته باشند و شماره تلفن z وجود داشته باشد که با x و y نسبت $i - 1$ ام داشته باشد. برنامه باید

- یک تماس جدید را ثبت کند.
- شماره تلفن x و y را از کاربر گرفته و مشخص می کند که x و y آیا نسبتی با هم دارند یا نه و اگر نسبتی دارند از کدام درجه است.

۱۱. بازی زندگی یک جدول دو بعدی با ابعاد $n \times n$ را فرض بگیرید. فرض کنید $L(i, j)$ خانه i, j ام جدول باشد. در این بازی به هر خانه از جدول یک سلول گفته می شود. هر سلول دو حالت می تواند داشته باشد: زنده یا مرده. سلولهای مجاور همسایه هستند. فرض بر این است که ستونهای انتهایی چپ و راست و سطرها بالایی و پایینی جدول مجاور هستند. پس هر سلول جدول هشت همسایه دارد.

			★	$L(1,5)$	★		
			★	★	★		
●	●	●					
●	$L(3,2)$	●					
●	●	●					
			★	★	★		

شکل ۱: همسایه های خانه $L(1,5)$ با رنگ قرمز مشخص شده اند و همسایه های خانه $L(3,2)$ با رنگ آبی.

قوانین زیر در مورد بازی جدول حاکم است.

- هر سلول زنده با کمتر از ۲ همسایه زنده، می میرد. (به دلیل کمبود جمعیت)
- هر سلول زنده با بیش از ۳ همسایه زنده، می میرد. (به دلیل ازدحام جمعیت)
- هر سلول زنده با ۲ یا ۳ همسایه زنده، زنده می ماند و به نسل بعد می رود.
- هر سلول مرده با دقیقاً ۳ همسایه زنده، دوباره زنده می شود.

می خواهیم بدانیم با اگر با وضعیت خاصی شروع کنیم، در نهایت وضعیت جدول به چه صورت خواهد بود. برنامه باید تغییر وضعیت ها را بصورت یک انیمیشن نشان دهد.

۱۲. سامانه فروشگاه در این پروژه هدف طراحی یک سامانه کامپیوتری برای ثبت تراکنشهای یک فروشگاه با وسایل الکترونیکی است که بر اساس طراحی شی گرا انجام شده است. برای سادگی فروشنده با تنها یک شعبه را در نظر می گیریم. برنامه باید امکانات زیر را ارائه دهد. اطلاعات باید در قالب فایل های متنی ذخیره شوند.

- لیست موجودی: باید لیستی از اقلام موجود (در فروشگاه و انبار) را داشته باشد. هر قلم کالا شامل اطلاعاتی مانند تعداد، موجودی در انبار، موجود در فروشگاه، قیمت خرید (برای یک واحد)، قیمت فروش (برای یک واحد) است.
- انتقال کالا: در صورت انتقال کالا از انبار به فروشگاه (یا فروشگاه به انبار) باید اطلاعات مربوط به انتقال در سامان منعکس شود.
- فروش کالا: در صورت فروش باید اطلاعات مربوط به فروش ثبت شود (چه مقدار کالا و به چه قیمتی به فروش رفته است به همراه تاریخ فروش و اسم خریدار).
- خرید کالا: اضافه کردن به انبار و ثبت اطلاعات مربوط به کالاهای خریداری شده.
- گزارش از میزان خرید و فروش ماهانه و سود عاید شده.