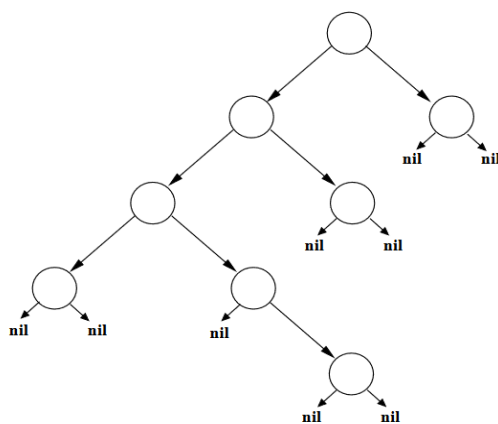
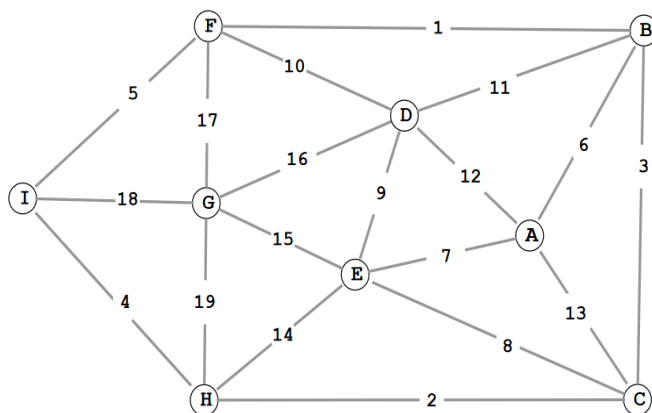
 دانشگاه علم و صنعت ایران	۱۴۰۰	درس: ساختمان داده‌ها	به نام او
	نام و نام خانوادگی: شماره دانشجویی:		
نمره	نکات: <ul style="list-style-type: none"> • لطفا شماره دانشجویی، نام و نام خانوادگی خود را در تمامی صفحات بنویسید. • درک صورت مسئله، بخشی از سوال است. بنابراین درک خود از مسئله را با استاد درس دابل چک نفرمایید. 		
۲.۵	بدترین زمان اجرای $f(n)$ را برای توابع زیر بنویسید. a) Insertion sort, Heapsort, Prim's b) $f(n) = n^2 \log n^2 + 2n \lg^2 n$ c) $f(n) = (n(100n + 5 + n^3))^2$ d) $f(n) = f(n/3) + f(n/6) + \theta(n^{\sqrt{\lg n}})$		
۲	با ذکر دلیل صحیح و یا غلط بودن موارد زیر را اثبات کنید (توضیحات به صورت کامل ارائه گردد): الف) فرض کنید از یک تابع درهم‌سازی h (hash function) برای درهم‌سازی n کلید متمایز در آرایه T به طول m ، استفاده می‌کنیم. با فرض درهم‌سازی یکنواخت ساده (simple uniform hashing)، تعداد مورد انتظار از برخورد جفت عناصرها (colliding) برابر با $\theta(n^2/m)$ است. ب) هر شبکه‌ی مرتب‌سازی با n ورودی، دارای عمق $\Omega(\lg n)$ است. ت) هیپ سورت، می‌تواند به عنوان روال مرتب‌سازی کمکی در مرتب‌سازی radix استفاده گردد، زیرا در محل عمل می‌کند (in place).		
۲.۵	ا) کلیدهای زیر را (شروع از چپ) به گره‌های درخت جستجوی دودویی (binary search tree) زیر اختصاص دهید به صورتی که خاصیت درخت جستجوی دودویی را برآورده کنند. $2, 3, 5, 7, 11, 13, 17, 19$		



(ب) توضیح دهید که چرا این درخت جستجوی دودویی نمی‌تواند مطابق با قوانین یک درخت قرمز-سیاه (red-black tree) رنگ شود.

(ج) درخت جستجوی دودویی را می‌توان با انجام یک چرخش (rotation) به یک درخت قرمز-سیاه تبدیل کرد. درخت قرمز-سیاه حاصل را رسم کنید و هر گره را با «قرمز» یا «سیاه» برچسب‌گذاری کنید. کلیدهای قسمت (الف) را وارد کنید.

نمودار گراف وزن دار (edge-weighted) زیر را با ۹ رأس و ۱۹ یال در نظر بگیرید. توجه داشته باشید که وزن یال‌ها اعداد صحیح متمایز بین ۱ و ۱۹ هستند.



ا) دنباله یال‌ها را در درخت پوشای کمینه (MST) به ترتیبی که الگوریتم Kruskal شامل آن‌ها شود کامل کنید (نام یال‌ها را وزن آن‌ها در نظر بگیرید).

1 -----

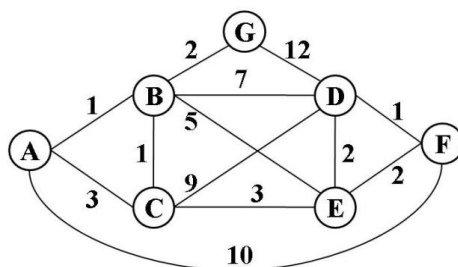
(ب) دنباله یال‌ها را در MST به ترتیبی که الگوریتم Prim شامل آن‌ها شود کامل کنید (نام یال‌ها را وزن آن‌ها در نظر بگیرید). الگوریتم Prim را از رأس A شروع کنید.

6 - - - - -

۲	<p>یک رأس پوششی گراف بدون جهت G مجموعه‌ای از رؤوس S است، به طوری که هر یال در G حداقل یکی از نقاط پایانی خود را در S دارد. مشکل رأس پوششی این است که: با توجه به یک گراف بدون جهت G، رأس پوششی G را پیدا کنید که دارای کوچک‌ترین اندازه ممکن باشد. یعنی تا حد امکان مجموعه‌ای از محدود رؤوسی را پیدا کنید که هر یال حداقل به یکی از رؤوس این مجموعه برخورد کند. در زیر یک الگوریتم پیشنهادی برای مسئله رأس پوششی ارائه شده است. این الگوریتم از مفهوم درجه‌ی یک رأس استفاده می‌کند، که تعداد یال‌هایی است که به این رأس برخورد می‌کنند.</p> <ol style="list-style-type: none"> 1) $S = \text{empty set.}$ 2) While G has at least one edge 3) Find a vertex v of the highest possible degree in G 4) Add v to S. 5) Remove vertex v from G, along with all edges that hit v 6) Answer S. <p>یک رأس پوششی باید تمام یال‌ها را در جایی که یک یال با انتخاب یک رأس از آن یال پوشانده می‌شود، پوشش دهد. این الگوریتم بر این اصل استوار است که با انتخاب رأسی که بیش‌ترین یال‌ها به آن برخورد می‌کند، بیش‌ترین یال‌ها را پوشش دهیم.</p> <p>این چه نوع الگوریتمی است؟ آیا پیدا کردن کوچک‌ترین رأس پوششی تضمینی است؟ اگر چنین است، چرا؟ اگر نه، مثال نقض بیاورید.</p>	۵
۳	<p>مشکل زیر را در نظر بگیرید:</p> <p>شما در حال نگهداری از n کودک هستید و $m > n$ کلوچه دارید تا بین آن‌ها تقسیم کنید. شما باید به هر کودک دقیقاً یک کلوچه بدهید (البته نمی‌توانید یک کلوچه را به دو کودک متفاوت بدهید). هر کودک دارای یک عامل حریصانه (greed factor)، G_i است ($n \geq i \geq 1$) که حداقل اندازه یک کلوچه است که کودک از آن راضی خواهد بود. و هر کلوچه دارای اندازه j ($m \geq j \geq 1$) است. هدف شما این است که تعداد فرزندان راضی را به حداکثر برسانید، به عنوان مثال، کودکانی که من یک کلوچه j با $G_i \geq j$ به آن‌ها اختصاص دادم.</p> <p>أ) شبه‌کد یک الگوریتم حریصانه (greedy algorithm) را برای حل مشکل واگذاری کلوچه بنویسید.</p> <p>ب) پیچیدگی الگوریتم خود را تعریف کنید.</p> <p>ج) آیا الگوریتم بهینه است؟ آن را ثابت کنید یا یک مثال نقض بیاورید.</p>	۶

مسیر	هزینه	رأس شناخته شده	رأس
			A
			B
			C
			D
			E
			F
			G

نمودار گراف وزن دار و بدون جهت زیر را در نظر بگیرید:

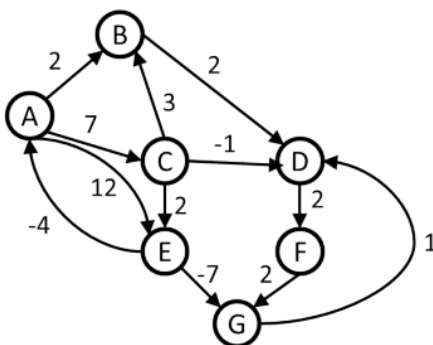


برای محاسبه‌ی کوتاه‌ترین مسیرهای تک منبعی (single-source) از A به هر رأس دیگر، از الگوریتم Dijkstra استفاده کنید. مراحل خود را در جدول زیر نشان دهید. در ادامه الگوریتم، مقادیر قدیمی را خط بزنید و مقادیر جدید را از چپ به راست در هر سلول بنویسید. همچنین رئوس را به ترتیبی که آن‌ها را مشخص کرده‌اید، فهرست کنید. در نهایت، کم‌هزینه‌ترین مسیر را از گره A به گره F نشان دهید.

رأس شناخته شده: -----

کم‌هزینه‌ترین مسیر از A تا F: -----

نمودار گراف وزن دار و جهت دار زیر را در نظر بگیرید:



<p>أ) با وجود اینکه نمودار دارای یال با وزن منفی است، از الگوریتم Dijkstra برای محاسبه‌ی کوتاه‌ترین مسیرها از A به هر رأس دیگر استفاده کنید. مراحل خود را در جدول زیر نشان دهید. در ادامه الگوریتم، مقادیر قدیمی را خط بزنید و مقادیر جدید را از چپ به راست در هر سلول بنویسید. همچنین رئوس را به ترتیبی که آن‌ها را مشخص کرده‌اید فهرست کنید.</p> <p>رأس شناخته‌شده: -----</p>			
مسیر	هزینه	رأس شناخته‌شده	رأس
			A
			B
			C
			D
			E
			F
			G

| ب) الگوریتم Dijkstra مسیر اشتباهی را برای برخی از رئوس پیدا کرد. فقط برای رأس‌هایی که مسیر اشتباه محاسبه شده است، هم مسیر محاسبه شده و هم مسیر صحیح را نشان دهید. ج) کدام یال را می‌توان از نمودار حذف کرد به‌طوری‌که الگوریتم Dijkstra پاسخ‌های صحیح را برای همه رئوس در نمودار باقی‌مانده محاسبه کند؟ | | | |
| ۴.۵ | پاسخ به این سوال اجباری نیست و نمره‌ی آن به صورت اضافه‌تر از نمره‌ی پایان‌ترم در نظر گرفته می‌شود: فرض کنید شما برای مدیریت فرآیند ترجمه برای برخی اسناد استخدام شده‌اید. متأسفانه، بخش‌های مختلف اسناد به زبان‌های مختلف نوشته شده است: در مجموع n زبان. رئیس شما می‌خواهد که کل اسناد به همه‌ی n زبان موجود باشد. m مترجم مختلف برای استخدام وجود دارد. برخی از این مترجمان داوطلب هستند که برای خدمات خود پولی دریافت نکنند. هر مترجمی دقیقاً دو زبان مختلف را می‌داند و می‌تواند هر دو آن زبان‌ها را ترجمه کند. هر مترجم هزینه استخدام غیرمنفی دارد (برخی ممکن است رایگان کار کنند). متأسفانه، بودجه شما برای استخدام یک مترجم برای هر جفت زبان بسیار کم است. در عوض، باید به زنجیره‌ای از مترجمان تکیه کنید: یک مترجم انگلیسی-اسپانیایی و یک مترجم اسپانیایی-فرانسوی که با هم کار می‌کنند، می‌توانند بین انگلیسی و فرانسوی ترجمه کنند. هدف شما یافتن مجموعه‌ای از مترجمان با حداقل هزینه است که به شما امکان می‌دهد بین هر جفت زبان ترجمه کنید. می‌توان این مسئله را به صورت یک گراف متصل بدون جهت $G = (V, E)$ با وزن یال‌های | | | ۹ |

	<p>غیر منفی w (یعنی صفر یا مثبت) فرموله کنیم. رئوس V زبان‌هایی هستند که می‌خواهید برای آن‌ها ترجمه ایجاد کنید. یال‌های E مترجم‌ها هستند. وزن یال $w(e)$ برای مترجم e هزینه استخدام مترجم $w(e)$ را نشان می‌دهد. یک زیرمجموعه $S \subseteq E$ از مترجمان می‌تواند برای ترجمه بین $a, b \in V$ استفاده شود اگر و فقط اگر زیرگراف $G_S = (V, S)$ حاوی مسیری بین a و b باشد. مجموعه $S \subseteq E$ یک شبکه ترجمه است اگر و فقط اگر S بتواند برای ترجمه بین تمام جفت‌های $a, b \in V$ استفاده گردد.</p> <p>ا) ثابت کنید که هر درخت پوشا حداقلی G (minimum spanning tree) نیز یک شبکه ترجمه با حداقل هزینه است.</p> <p>ب) مثالی از یک شبکه‌ی ترجمه با حداقل هزینه بنزید که درخت پوشا حداقلی G نباشد.</p> <p>ج) یک الگوریتم کارآمد ارائه دهید که G را به عنوان ورودی دریافت کند و یک شبکه ترجمه با حداقل هزینه G را به عنوان خروجی دهد. زمان اجرای الگوریتم خود را بر حسب تعداد زبان n و تعداد مترجمان بالقوه m بیان کنید.</p>	
--	--	--

با آرزوی بهترین روزگار

ح.رحمانی