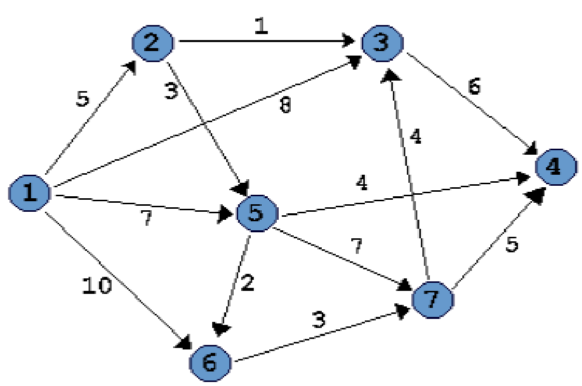
جواب تکلیف دوم

طراحی الگوریتم
دانشکده ریاضی. دانشگاه صنعتی خواجه نصیرالدین طوسی. پاییز ۱۴۰۳

۱. الگوریتم دایکسترا برای گراف $G = (V, E)$ و راس مبدا s، با استفاده از صف اولویت pq بصورت زیر پیاده‌سازی شده است. اینجا $w(e)$ طول یال e را نشان می‌دهد.

```
DIJKSTRA (V, E, w, s)
--------------------------------------------------
d[s] = 0
FOREACH v in V/s: d[v] = ∞
Create an empty priority queue pq
FOREACH v in V: INSERT(pq, v, d[v])
WHILE (IS-NOT-EMPTY(pq))
   u = DEL-MIN(pq)
   FOREACH edge e = (u, v) in E leaving u:
      IF d[v] > d[u] + w(e):
         DECREASE-KEY(pq, v, d[u] + w(e))
         d[v] = d[u] + w(e)
```

با شروع از راس 1 در گراف زیر الگوریتم بالا چند بار عمل DECREASE-KEY را انجام می‌دهد؟



۱۰ بار عمل decrease-key انجام می‌شود. موقع پردازش راس ۱، به تعداد ٤ بار، موقع پردازش ۲ به تعداد ۱ بار، موقع پردازش راس ۳ به تعداد ۱ بار، موقع پردازش راس ۵ به تعداد ۳ بار و موقع پردازش راس ٦ به تعداد ۱ بار.

۲. با توجه به جدول زیر، برای مجموعه کاراکتر $\{A, B, C, D, E\}$ یک کد هافمن طراحی کنید.

character	A	B	C	D	E
probability	0.4	0.1	0.2	0.15	0.15

یک جواب می‌تواند بصورت زیر باشد.

$$A = 1, B = 000, C = 010, D = 001, E = 011$$

۳. کدام از یک کدهای زیر نمی‌تواند یک کد هافمن باشد. چرا؟

(آ) $\{0, 10, 11\}$

(ب) $\{00, 01, 10, 110\}$

کد بهینه نیست چون کد 110 را می‌توان کوتاهتر کرد و پیشوندی بودن را حفظ کرد. لذا نمی‌تواند کد هافمن باشد.

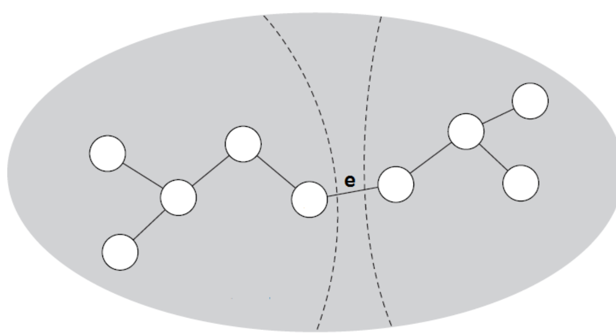
(ج) $\{0, 01, 11\}$

کد پیشوندی نیست. پس نمی‌تواند کد هافمن باشد.

۴. نشان دهید اگر وزن یالها منحصر بفرد باشد، درخت فراگیر کمینه نیز منحصر بفرد است.

فرض کنید دو درخت کمینه فراگیر T_1 و T_2 برای گراف G وجود دارد که متمایز هستند.

یال $e \in T_1$ را در نظر بگیرید که در درخت T_2 حضور ندارد. برداشتن یال e درخت T_1 را به دو مولفه همبند تقسیم میکند.



با توجه به قضیه ای که در کلاس اثبات کردیم، چون وزن یالها متمایز هستند و T_1 یک درخت فراگیر کمینه است پس e باید سبکترین یال در برش $(S, V \setminus S)$ باشد. لذا e حتما باید در درخت فراگیر کمینه حضور داشته باشد. این با کمینه بودن T_2 در تناقض است. لذا T_1 و T_2 باید یکسان باشند.

۵. نشان دهید الگوریتمی که بر اساس حذف یالها (برعکس کروسکال) عمل می‌کند، درخت فراگیر کمینه را پیدا می‌کند. می‌توانید فرض کنید که وزن یالها منحصر بفرد است.

نشان می‌دهیم یالی که حذف می‌شود حتما سنگین ترین یال یک دور است و لذا طبق لمی که در کلاس ارائه شد نمی‌تواند جزو درخت فراگیر کمینه باشد. چون الگوریتم در نهایت یک درخت فراگیر تولید می‌کند، یالهایی که باقی می‌مانند باید عضو درخت فراگیر کمینه باشد.

دقت کنید یال e که حذف می‌شود حتما باید عضو یک دور باید باشد چون در غیر اینصورت حذف آن خسارت جبران ناپذیری به همبندی گراف وارد خواهد کرد. علاوه بر این یال e باید سنگینترین یال هر دوری باشد که در آن شرکت دارد. فرض کنید $e \in C$ وقتی که C یک دور است. دقت کنید موقع بررسی e یالهای دیگر C هنوز بررسی نشده‌اند چون حذف آنها همبندی گراف را نقض نمی‌کرد. لذا e باید قبل از یالهای C بررسی شده باشد و طبق فرض ما و منطق الگوریتم باید سنگینترین یال دور باشد.

۶. گراف جهت دار $G(V,E)$ داده شده است. می‌خواهیم پیامی را از راس s به راس t منتقل کنیم. برای ارسال پیام، مسیری از s به t انتخاب می‌شود. برای هر یال (u,v) در گراف احتمال $p(u,v) \in [0,1]$ داده شده است. احتمال $p(u,v)$ به این معنی است که چقدر احتمال دارد که پیام مخابره شده از u به v برسد. می‌خواهیم مسیر L از s به t را پیدا کنیم که حاصلضرب

$$\prod_{(u,v)\in L} p(u,v)$$

را ماکزیمم کند. این حاصلضرب برابر با احتمال رسیدن پیام از s به t از طریق مسیر L است. یک راه حل برای این مسئله پیشنهاد دهید. چون $\log$ یک تابع صعودی است، داریم

$$\text{argmax}_L \prod_{(u,v)\in L} p(u,v) = \text{argmax}_L \ \log \prod_{(u,v)\in L} p(u,v)$$

از طرف دیگر $\text{argmax} f(x) = -\text{argmin} f(x)$ لذا داریم

$$\text{argmax}_L \ \log \prod_{(u,v)\in L} p(u,v) = -\text{argmin}_L \ \log \prod_{(u,v)\in L} p(u,v) = \text{argmin}_L \ -\sum_{(u,v)\in L} \log p(u,v)$$

لذا کافی است که مسیر L را پیدا کنیم که تابع هدف زیر را کمینه می‌کند.

$$\sum_{(u,v)\in L} \log p(u,v)^{-1}$$

پس وزن یال (u,v) را برابر با $\log p(u,v)^{-1}$ قرار می‌دهیم و کوتاهترین مسیر را بین s و t با استفاده از الگوریتم دایکسترا پیدا می‌کنیم.

۷. می‌دانیم که اختلاف قد می‌تواند یک عامل تاثیرگذار در مسابقات ورزشی باشد (شاید تنها دلیل باخت تکواندوکار ایرانی در مقابل حریف مراکشی در مسابقه فینال المپیک مسئله اختلاف قد بود!) حال دو گروه ورزشکار

$$A = \{a_1, \cdots, a_n\} \text{ و } B = \{b_1, \cdots, b_n\}$$

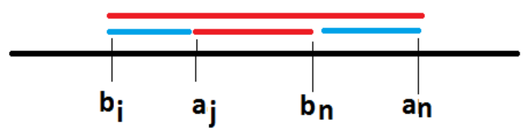
داریم که می‌خواهیم دو به دو مسابقه دهند. اگر تابع $h()$ قد یک فرد را مشخص کند، می‌خواهیم هر ورزشکار از گروه A با یک ورزشکار از گروه B متناظر کنیم بطوریکه

$$\sum_{i=1}^{n} (|h(a_i) - h(\sigma(a_i))|$$

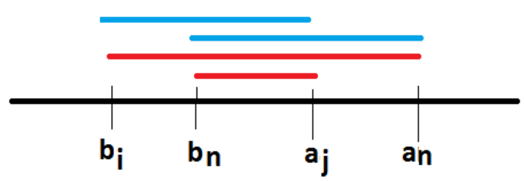
کمینه شود. ایجا $\sigma : A \to B$ تناظر یک به یک بین A و B را مشخص می‌کند. برای این مسئله یک الگوریتم حریصانه پیشنهاد دهید که مجموع اختلاف قد ها را کمینه می‌کند.

استراتژی حریصانه: بلند قد ترین فرد در مجموعه A را با بلند قدترین فرد در مجموعه B در یک گروه دوتایی قرار می‌دهیم. به همین طریق ادامه می‌دهیم و از میان افراد باقیمانده گروه های دوتایی را با ملاکی که گفتیم می‌سازیم. به عبارت دیگر فرض کنید $a_1, \cdots, a_n$ از کوتاه به بلند مرتب شده است. برای $b_1, \cdots, b_n$ هم همینطور. ادعا می‌کنیم تناظر $(a_1, b_1), \cdots, (a_n, b_n)$ جواب مورد نظر است. فرض کنید در تناظر بهینه دو فرد a_n و b_n با هم نباشند. پس حتما b_i وجود دارد که با a_n متناظر شده و b_n هم با یک a_j متناظر شده. ادعا می‌کنیم اگر تناظر را تغییر دهیم و a_n را کنار b_n قرار دهیم و b_i را با a_j متناظر کنیم جمع اختلاف قدها بدتر نمی‌شود. بدون کاهش از عمومیت مسئله فرض کنید $h(a_n) \geq h(b_n)$. دو حالت داریم:

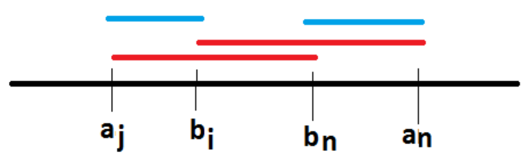
- $h(b_i) \leq h(a_j) \leq h(b_n) \leq h(a_n)$. در این حالت اگر تعویض مورد نظر را انجام دهیم قطعا وضعیت بهتر خواهد شد. به شکل زیر توجه کنید. قطعات آبی طول کمتری دارند.



- $h(b_i) \leq h(b_n) \leq h(a_j) \leq h(a_n)$. در این حالت اگر تعویض مورد نظر را انجام دهیم وضعیت بدتر نخواهد شد. به شکل زیر توجه کنید. قطعات آبی و قرمز طول یکسان دارند.



- $h(a_j) \leq h(b_i) \leq h(b_n) \leq h(a_n)$. در این حالت اگر تعویض مورد نظر را انجام دهیم قطعا وضعیت بهتر خواهد شد. به شکل زیر توجه کنید. قطعات آبی طول کمتری دارند.



پس در یک تناظر بهینه زوج a_n و b_n با هم هستند. می‌توان همین استدلال را برای a_{n-1} و b_{n-1} و بقیه لیست بکار برد.

۸. مریم خیلی مهمانی دوست دارد و وضع مالی‌اش هم خوب است. او به این فکر می‌کند که چه افرادی را به مهمانی اش دعوت کند. او می‌خواهد مهمانی جوری باشد که افراد احساس غریبی نکنند و عین حال برای هم خیلی هم تکراری نباشند. برای همین او این قانون را وضع می‌کند که هر فردی که دعوت می‌شود باید حداقل ۴ فرد دیگر (غیر از میزبان) را بشناسد و حداقل ۴ نفر هم در مهمانی باشند که نمی‌شناسد. هر فردی که دعوت می‌شود میزبان را می‌شناسد. فرض کنید که مریم از دایره آشنایانش می‌داند هر فرد دقیقا چه کسانی را می‌شناسد (شناختن اینجا یک رابطه دو طرفه است.) با این قانونی که مریم وضع کرده است، کمک کنید که او بیشترین تعداد افراد را به مهمانی اش دعوت کند.

کافی است که گرافی از دوستان مریم تشکیل دهیم بدین ترتیب که رئوس دوستان مریم هستند و یالها رابطه آشنایی هستند. راس با درجه کمتر از 4 را می‌توانیم با خیال راحت حذف کنیم. همینطور راس با درجه بیشتر از $n-4$ را وقتی که n تعداد رئوس در گرافی است که باقی می‌ماند. هر فردی که حذف می‌شود قطعا نمی‌تواند جزو افراد دعوت شده باشد. لذا آنانکه بعد از این پروسه قلع و قمح جان سالم بدر برده‌اند جواب مسئله هستند. این بیشترین تعداد ممکن است.