تمرینات سری ۳.

الگوریتم های پیشرفته.

سؤال ۱) روش حل به این گونه هست که ما تا جایی که می توانیم جلو می رویم و سوخت مصرف می کنیم و در ایستگاهی که سوختمان به ایستگاه بعدی نمی رسید سوخت گیری می کنیم.

اثبات درستی: اسم سری به وجود آمده از ایستگاه های سوخت گیری را G می گذاریم فرض کنید حالت G' حالت بهینه ما باشد. این ۲ الگوریتم را با هم مقایسه می کنیم تا به اولین نقطهٔ تفاوت برسیم و اسم آن را A می گذاریم می دانیم هر ۲ روش G و G' در ایستگاه های قبل از A کاملاً مثل هم هستند پس G' بعد از ایستگاه A به ایستگاه دیگری مثل B' رفته و G به ایستگاهی مانند B رفته است که $B \neq B'$ چون B دورترین ایستگاه ممکن به A بوده است (چون الگوریتم این طور انتخاب می کند!) پس B از B' جلوتر است حالا در ایستگاه بعد از B و B' به ایستگاه های B_1 و B'_1 خواهند رفت که آنها هم یا برابرند که تا جایی که برابرند جلو می رویم یا به مقصد می رسیم که تعداد ایستگاه های ایستاده شده هر ۲ آنها برابر است که با فرض بهینه بودن G' در تناقض است و یا به ۲ ایستگاه دیگر مثل ... می رسند که آن ۲ ایستگاه مثل B و B' است و با همان استدلال B و B' می توان همچنان ادامه داد. یا $B_1 \neq B'_1$ که با استدلالی مثل B و B'، B'_1 از B_1 عقب تر است که اگر همینطور جلو برویم G زودتر و با تعداد کمتری توقف به مقصد می رسد و با بهینه تر بودن G' در تناقض است پس در نتیجه این الگوریتم همواره درست بهینه را می یابد

کدی برای الگوریتم:

```
find min (d[n], m)
P = m
i = 0
Count = 0
for (i = 0  to n)
    if (P + d[i] - d[i+1] > 0)
        P = P + d[i] - d[i+1]
    else
        P = m
        Count ++
return  count
```

همان طور که می بینید این مسئله از $\Theta(n)$ است.

روش دیگر:

ابتدا این سؤال را می توانست با پیچیدگی بیشتر با روش تقسیم و غلبه نیز حل کرد که همهٔ حالت های ممکن برای توقف اول را به دست آوردیم و آن بعدی را به ... حل حالت های قبل به دست آوردیم و حالت بعد آن موقعی است که ما تا ... حل $m-1$ مسئله داشته باشیم که d_1 تا d_m با ... باشند.

$$T(n) = T(n-1) + T(n-2) + \dots + T(n-m) + m \rightarrow$$ برای پیدا کردن min این $T(n-1)$ ها

سؤال ۲) اگر ما n چادر داشته باشیم بین آنها n-1 فاصله می‌افتد چون فاصله بین اولین و آخرین سقف خاکستری متوالی ثابت است و طول چادر برابر با تفاضل میان فاصله اولین و آخرین سقف خاکستری با مجموع فاصله‌های خالی میان چادرهاست هنگامی ما کمترین طول چادر را حساب می‌کنیم که بتوانیم بیش‌ترین مجموع فاصله‌های میان چادرها را بیابیم چون اول و آخر هر چادر یک سقف خاکستری است (هدف min کردن طول چادر) پس ما باید لیستی از فاصله میان هر ۲ سقف خاکستری متوالی را داشته باشیم و از میان آنها n-1 تا از بیش‌ترین اعداد را انتخاب کنیم و تنها برای این فضاهای خالی چادر در نظر نگیریم و بقیه را با چادر بپوشانیم

الگوریتم:

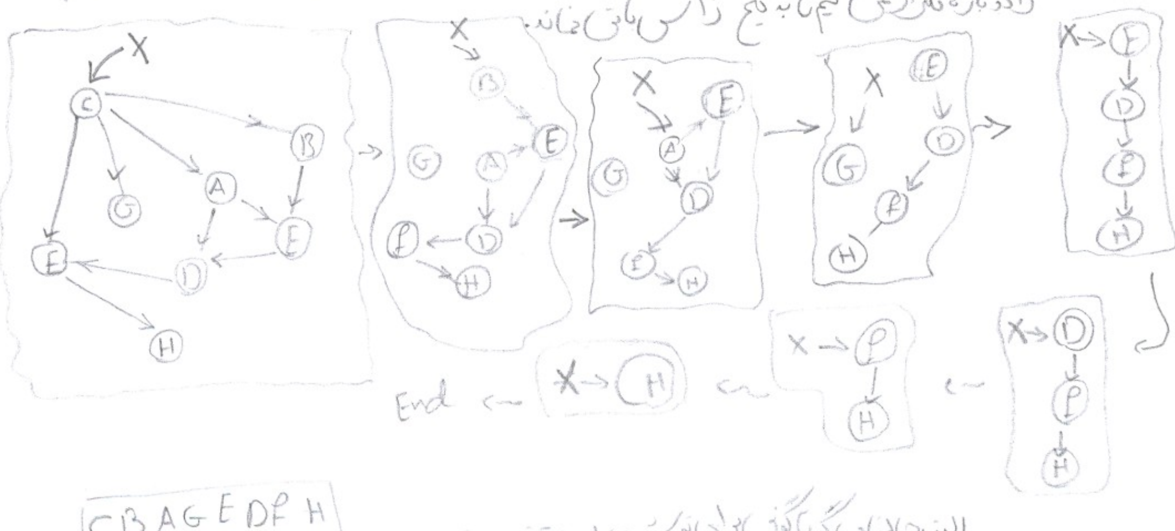
۱- پیدا کردن فاصله میان تمام سقف‌های متوالی خاکستری

۲- انتخاب n-1 تا از بزرگترین اعداد انتخاب قسمت ۱

۳- از اولین سقف خاکستری شروع می‌کنیم تا به اولین سقف انتخابی در ۲ برسیم و به همه تا سقف‌های چادر را قطع می‌کنیم و در سقف خاکستری بعدی دوباره آن را ادامه می‌دهیم تا به اعداد دیگر ۲ برسیم این کار را تا آخر ادامه می‌دهیم.

سؤال ۳) اول باید دقت کرد که در این گراف‌ها نمی‌توانیم دور داشته باشیم چون اگر دوری موجود باشد، آنگاه به یکسری کارها می‌رسیم که همگی به دنبال اجرا شدن دیگری آغاز نشود و هیچ کاری نمی‌تواند آغاز شود.

با فرض بالا نتیجه می‌شود که رأسی وجود دارد که درجه ورودی‌اش صفر است ما اول همه آنها را پیدا می‌کنیم یکی از آنها را انتخاب می‌کنیم و بعد آن رأس و یال‌های مربوط به آن را حذف می‌کنیم و به علت عدم وجود دور در این گراف دوباره رأس‌هایی هستند که درجه ورودی‌شان ۰ باشد آنها را دوباره انتخاب می‌کنیم و قسمت‌های قبل را دوباره تکرار می‌کنیم تا به هیچ رأسی باقی نماند.



C B A G E D F H

البته الگوریتم‌های گوناگونی برای نوشتن این ترتیب موجود است که بستگی به انتخاب کدام یک از رأس‌ها با درجه ورودی صفر دارد.

جواب سؤال ۴: چند قانون داریم که در هر الگوریتمی برای این مسئله باید رعایت شود.

۱- هر رأسی می‌آید که پدرهایش حاضر باشد.

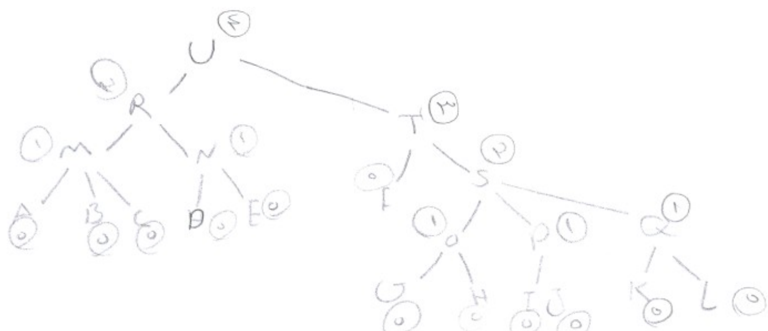
۲- رأسی که تمام بچه‌هایش آمده، نشان می‌توان حذف شود (چون اگر نباشد آنگاه باید برای بچه‌اش بنابه قانون ۱ بیاید که شرط مسئله را به هم می‌زند)

۳- حداکثر ما می‌توانیم ۵ رأس داشته باشیم.

بنابه قانون ۲ و ۳ می‌توانیم بگوییم که حداکثر ما می‌توانیم تا ۴ نسل جلوتر برویم.

الگوریتم: روی هر رأسی بیشترین فاصله تا برگی که آن جزئش هست را می‌نویسیم و بعد اول ریشه را انتخاب می‌کنیم و بعد از میان فرزندان آن، آن که کوچکترین عدد را دارد را انتخاب می‌کنیم تا همه‌ی بچه‌های یک رأس را انتخاب کنیم آن رأس را حذف می‌کنیم و به پایین می‌رویم.

اثبات: فرض کنید که نتوانیم که درستی را اثبات کنیم آنگاه فرض می‌کنیم برابر باشیم در فاصله‌ها ظاهری شونده‌های حداکثر نیست چون ما برای حذف آنها باید تا برگ جلو برویم ولی با یک فرض کمتر از ۵ چون برادرش هم جای آن را گرفته. پس اگر ما با یک گراف بتوانیم مسئله را حل کنیم باید ۲ برادر نیاید.



→ $\boxed{U\ R\ N\ E\ D}$ → $\boxed{U\ R\ M\ A\ B}$ → $\boxed{U\ R\ M\ C\ T}$ → $\boxed{T\ F\ S\ Q\ L}$

$\boxed{S\ Q\ K\ P\ J}$ → $\boxed{S\ P\ I\ O\ G}$ → $\boxed{H}$