علیرضا کریمی 810101492

1) برای کاهش مسئله A به B، از آنجایی که تعداد حروف الفبای انگلیسی 26 تاست، پس به 5 بیت نیاز داریم. برای هر حرف، به ترتیب، ۵ رقم بیت معادل آن حرف را می‌گذاریم. مثلا داریم:

A → 00000 , B → 00001 , C → 00010 , ...

به این شکل در مسئله B، تعداد بیت‌ها در کلمه روی مهره و پشت مهره حتما مضرب 5 هستند. حال درستی راه خود را اثبات می‌کنیم.

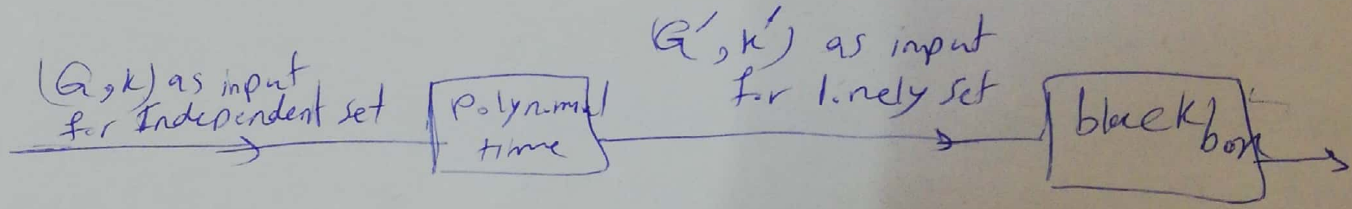
اگر در A به جواب بله رسیده باشیم، یعنی در بالا و پایین یک ایندکس‌ها به از حروف داریم (پس تعداد حروف بالا و پایین در کل برابر است). حال اگر به ازای هر حرف، 5 رقم بیت مربوط به آن را بگذاریم، چون ۵ رقم ۵ رقم عینا یکی هستن، پس باز هم در مسئله B، عبارت روی مهره‌ها با عبارت زیر مهره‌ها برابر می‌شود. اگر در B جواب داشته باشیم که بله باشد، از آنجایی که جواب‌ها در ما (عبارات بالای مهره و پایین مهره‌ها) مضرب ۵ هستن و کلمه‌های روی هر مهره و پشت هر مهره هم مضرب ۵ هستند، و چون یک یک بیت‌ها با هم سازگار شدن هم بالا رو هم پایین مهره‌ها، پس هر ۵ تا ۵ تا بیت هم سازگار هستن و پس اگر آنها را به حرف معادل شان هم تبدیل کنیم، آن حرف‌ها یکی خواهند بود، پس کلمات عبارت بالا و پایین با هم برابر می‌شن و مسئله A حل می‌شه.

این تولید کردن 5 رقم به ازای هر حرف هم از اردر ثابت است و اگر n حرف داشته باشیم 5n بیت تولید می‌شود که از اردر ثابت و چندجمله‌ای است (هزینه map کردن هم از اردر ثابت است.)

برای اثبات NP hard بودن، کافی است که یک مسئله NP complete را به آن کاهش دهیم.

۲) مسئله Independent Set را به مسئله راس‌های تنها کاهش خواهیم داد: Independent Set $\leq_p$ lonely set

ورودی مسئله Independent Set، یک $G=(V, E)$ و یک k است و دنبال این هستیم که چک کنیم آیا وجود دارد k راس‌ها که هیچ کدام از دو تا از آن k راس، به هم یال نداشته باشند. برای کاهش آن به lonely set، به ازای هر راس، یک راس جدید قرار می‌دهیم و یک یال از راس جدید به راس قدیم متناظرش وصل می‌کنیم. حال کافی است که گراف جدید را به همراه $k=k+|V|$ را به مسئله lonely set بدهیم و ببینیم آیا یک زیرگراف به اندازه $k+|V|$ راس که هر راس‌های آنها lonely باشند وجود دارد یا نه. ثابت خواهیم کرد که اگر وجود داشته باشد، یعنی یک زیرگراف k راسی در Independent set وجود دارد که هیچ کدام از راس‌های آن به هم یال ندارند.



$$G' = (E', V')$$

$$|E'| = |E| + |V|$$

$$|V'| = |V| + |V|$$

$$k' = k + |V|$$

همانطور که می‌بینید، ساخت ورودی جدید برای مسئله lonely set از روی ورودی Independent set از اردر $O(|V|)$ یعنی از اردر چندجمله‌ای است.

حال دو طرف قضیه را ثابت می‌کنیم. درصورتی که یک زیرگراف به اندازه k راس در گراف G داشته باشیم که هیچ کدام از آنها دو به دو به هم یال نداشته باشند، با اضافه کردن $|V|$ راس جدید به گراف G که هر کدام تنها یک یال دارند، پس $k+|V|$ راس در گراف جدید G' وجود دارند که تنها یک یال به راس‌های این زیرگراف دارند. (چون راس‌های جدید که حداکثر 1 یال دارند k راس‌های قدیمی هم حالا با اضافه شدن راس جدید متناظر آنها حداکثر 1 یال دارند) پس حتما $k+|V|$ راس وجود دارند که بدون یال درون زیرگراف درون زیرگراف lonely set باشند. پس $k+|V|$ جواب بله می‌دهد که lonely set

ادامه سوال 2) حال اگر $|V|+k$ راس از راس های گراف G'، یک lonely set تشکیل بدهند، از آنجایی که G حداکثر $|V|$ تا راس داشت یعنی از میان این $|V|+k$ راسی که lonely set را تشکیل داده اند، قطعا حداقل k تا از آنها از میان راس های جدید است. حداکثر هم $|V|$ تا از آنها از میان راس های جدید خواهد بود. حال به ازای هر دو راس قدیمی که اگر به هم یال داشته باشند، یعنی راس های جدید متناظر به آنها را انتخاب نکردیم. به ازای دو سر هر یال درون راس های قدیمی، به جای یکی از راس های قدیمی، می توانیم آن را برداریم و راس جدید متناظر به آن را قرار دهیم، الان هنوز هم به همان تعداد در زیرگراف داریم و حتی یک یال بین راس های قدیمی را هم برداشتیم. (دقت کنید اگر دو راس قدیمی به هم یال داشته باشند قطعا راس های جدید متناظر به آنها انتخاب نشده. با برداشتن این یال ها اضافه کردن (یعنی حذف یکی از آن راس های قدیمی) و اضافه کردن راس های جدید متناظر به آنها، می توان یال های بین راس های قدیمی را به این شکل حذف کرد و حتی به زیرگراف lonely set ساز هم اضافه نشود.) با ادامه این روند تمام یال های بین راس های قدیمی را می توان حذف کرد و هنوز هم جواب ما معتبر است (این کار از رتبه $O(E)$ و چند جمله ای هست). حال می دانیم قطعا حداقل k راس از $|V|+k$ راسی انتخاب شده از راس های قدیمی هستند و با تغییراتی که دادیم مطمئن هستیم که این k راس هیچ یالی به هم ندارند. پس اگر هم در گراف G به این k راس نگاه کنیم، هیچ یالی به هم نخواهند داشت و Independent Set تشکیل می دهند.

یک lonely set از سایز $|V|+k$ $\Longleftrightarrow$ یک Independent Set از سایز k

تمام تبدیل ها را هم که گفتیم از اردر چند جمله ای بود.

سوال 3) برای اثبات NP-hard بودن آن، مسئله vertex cover را به آن می‌خواهیم کاهش دهیم، اما قبل از آن باید ثابت کنیم که مسئله‌ی ما NP است که یعنی می‌توانیم در زمان چندجمله‌ای verify کنیم که یک جوابی که بهش دادیم درست هست یا نه؟ می‌دانیم که برای certificate، باید یه نوعی یک جواب مسئله را به آن بدهیم.

باید چک کنیم که اولاً تعداد دورها از اردر چندجمله‌ای و نه نمایی باشد (چون اگر بخواهد دورهایی که می‌خواهیم چک کنیم از اردر نمایی هم داشته باشه، یعنی یک سری از دورها رو داره چند بار حل می‌کنه و یعنی دورهای ما تکراری هستن، پس فقط باید از همان تعداد دورهای متمایز استفاده بشه که آن هم از اردر چندجمله‌ای هست. یعنی طول certificate ما نباید بیشتر از تعداد دورها باشه و باید از اردر ورودی (یعنی چندجمله‌ای باشه) بعد به ازای هر دور، پیمایش می‌کنیم (از اردر چندجمله‌ای پیمایش انجام میشه که حداکثر از اردر $O(E^2)$ میشه فرض کنیم که بازم چندجمله‌ای هست) بعد که همه رو پیمایش کرد و راس‌هایی که دید، visited می‌کنه. اگر در آخر همه راس‌ها visited شده بود جواب درسته و بله است در غیر اینصورت جواب خیر است. (ساخته شدن گراف جدید از همه از اردر چندجمله‌ای نسبت به ورودی است که در انتها توضیح می‌دم. ولی حداکثر $|E|$ تا راس و حداکثر $|V|$ دور داریم پس تعداد راس‌ها و یال‌ها و دورها از اردر چندجمله‌ای نسبت به ورودی خواهد شد) (تعداد k دور در certificate باید کمتر مساوی k داده شده در ورودی باشد)

اثبات NP-Hard بودن، از مسئله vertex cover به آن کاهش می‌دهیم. به طوری که به ازای تک‌تک یال‌ها در گراف G، یک راس نماینده آن یال در گراف G' می‌گذاریم. علاوه بر همه راس‌هایی که متناظر هر یال هستند و در G' اضافه شدند، راسی به نمایندگی از S در G' اضافه می‌کنیم.

حال به ازای تک‌تک راس‌های موجود در G، باید معادل آن یک دور در G' درست کنیم. این کار را به این شکل می‌کنیم که از S شروع می‌کنیم و به ازای تک‌تک یال‌های متصل به آن راس در G (فرض کنید آن راس v باشد و یال‌های e_1 و e_2 و ... و e_m به آن در G وصل باشند. به یاد دارید که به ازای هر یال در G، یک راس نماینده در G' گذاشتیم.) حال از S به راس نماینده e_1، از آن به راس نماینده e_2 و ... و در آخر از راس نماینده e_m به S در گراف G'، یال می‌زنیم. به ازای هر راس و یال‌های متصل به آن در G، یک دور در G' تشکیل دادیم.

(ادامه ۳) منظور این کار این است هر نظور که در مسئله vertex cover ، اگر یک راس را انتخاب کنیم ، همه راس های متصل به آن علامت زده می شن ، در گراف G' ، انتخاب این راس ، معادل پیمایش دور مربوط به آن راس است که با گذاشتن از تمام راس های نماینده یال های متصل به این راس در G

حال اگر بتوانیم مثلا k تا دور انتخاب کنیم و با این k دور تمام راس ها را ببینیم و پیمایش کنیم در گراف G'

این مثل این است که در گراف G ، k تا راس را رنگ زدیم و اینگونه همه یال ها یک سمتشان رنگی شده

اثبات درستی دوطرفه

اگر مسئله vertex cover ، با رنگ کردن k راس بتوانیم تمام یال ها را علامت بزنیم ، این معادل این است که آن k دور شامل تمام یال های متصل به آن راس های رنگ زده شده را انتخاب کنیم . و یال ها را هر کدام با راس نشان می دادیم در G' پس اون k راسی در G که تمام یال را رنگ می کرد ، k دور است که راس های هر دور نشان دهنده یال های متصل به آن راس های رنگ شده در G هستند در G'

پس با پیمایش هر کدام از k دور در G' ، تمام راس های نماینده آن یال های متصل دیده می شوند و چون همه یال ها را علامت زدیم ، با پیمایش این k دور همه راس های نماینده را پیمایش می کنیم و تمام پل ها خورده می شود . پس اون k دور جواب مسئله هموش و سیرها می شه

از طرف اگر بتوانیم با k دور تمام راس های گراف G' را پیمایش کنیم (همه پل ها خورده شه) می دانیم به ازای هر دور یک راسی وجود دارد که به ازای تک تک راس های این دور یال متناظر آن ، به آن وصل است ، k دور داریم . پس در G ، k راسی داشتیم که با انتخاب این k راس ، تمام یال های متصل آن در G علامت زده می شه . چون همه راس های در G' را دیدیم ، همه یال ها در G هم علامت زده می شه با انتخاب اون k راس متناظر اون دور ها . پس اون k راسی جواب vertex cover می شه .

آیا با k راسی می شه vertex cover را برقرار کرد $\Longleftrightarrow$ آیا با k دور می شه کل راس ها را دید

چون در G حداکثر $|V|$ تا راسی داریم پس در G' ساخته شده حداکثر $|V|$ دور داریم . چون هر راسی در G حداکثر $|V|-1$ تا یال وصل شده ، پس طول دور ها هم از اردر $|V|$ است . پس کل گراف ساخته شده G از اردر چند جمله ای ساخته می شه .

۴) در این مسئله از ما خواسته شده که NP-hard بودن آن را اثبات کنیم پس کافی است که یک مسئله NP-complete را در زمان چند جمله‌ای به آن کاهش دهیم. می‌توانیم از مسئله SAT (یا همان circuit SAT) کمک بگیریم

SAT $\leq_p$ Bola

به این شکل عمل می‌کنیم که هر ردیف ما، همان clause ما خواهد بود. و به ازای هر ستون هم یک متغیر می‌گذاریم. اگر که دانشجوی سالم در ستون i ردیف j بود، یعنی در clause C_j، x_i می‌گذاریم و اگر مریض بود، $\bar{x_i}$ می‌گذاریم (نقیض آن را). اگر که مسئله SAT به ما جواب بله بدهد، یعنی در هر clause حداقل یکی از متغیرها (چه خودش چه نقیض آن) درست بوده و اون clause true شده. این معادل شرط اول است که در هر ردیف حداقل یک نفر چه سالم چه مریض نشسته باشد چون هر clause نشان دهنده یک ردیف است. از طرفی داریم که مسئله SAT یکی از بدیهی‌ترین شروطش اینه که همزمان هم x_i هم $\bar{x_i}$ نمی‌تونن درست باشن این دقیقاً معادل شرط دوم است که در یک ستون (چون x_i نماینده اون ستونه) یا فقط مریض داریم یا سالم داریم. سازش هر clause هم از m (تعداد ستون‌ها) و تعداد clauseها هم از n (تعداد ردیف‌ها) است پس هزینه ساخت SAT و انجام این مدل سازی از $O(nm)$ که نسبت به ورودی چند جمله‌ای هستند، است. حال گفتیم که اگر SAT به ما جواب بله بدهد پس قطعاً مسئله Bola هم به ما جواب بله می‌دهد و شروط مسئله هم برقرار خواهد بود. حال باید طرف دیگر را بگوییم. یعنی اگر مسئله Bola به ما جواب بله بدهد، مسئله SAT هم به ما جواب بله می‌دهد یا خیر؟ اگر بتوانیم مسئله Bola را حل کنیم، یعنی در هر ستون یا فقط مریض داریم یا فقط سالم که یعنی یا x_i برقرار است یا $\bar{x_i}$. از طرفی اگر در هر ردیف هم حداقل یه دانشجو باشه یعنی اینکه اون ردیف شرطش برقراره که این معادل برقراری و true بودن clause متناظر با اون ردیف هست. پس مسئله SAT هم به ما بله می‌دهد اگر به ازای یک ردیف (یعنی clauseها) حداقل یک دانشجو وجود داشته باشه. می‌گوییم اگر که در اون ردیف i ستون j، دانشجو نبود، اصلاً متغیر x_j را در C_i clause نمی‌گذاریم که اصلاً یعنی آن را بررسی نمی‌کنیم.

ادامه سوال ۴) منظورمن سایز هر clause، حداکثر m است (چون ممکن است یک سری دانشجوها نباشن پس متغیر اونها هم نیست توی اون clauseها)

اگر در SAT هر یک سری متغیر در clause ها نبودن، یعنی اصلا دانشجوی آنها در اول ردیف و اول ستون حضور نداشته است.

پس داشتیم

حل مسئله SAT و گرفتن جواب بله $\Longleftrightarrow$ بله گرفتن از Bola و ممکن پذیر بودن چینش دانشجوها به شکل مطلوب (برقرار شروط)

همانطور که گفتیم این کاهش از اردر $O(\overset{\text{تعداد clause ها}}{n}\underset{\text{حداکثر سایز یک clause}}{m})$ است.

پس همانطور که دیدید، توانستیم به ازای هر متغیری که وجود دارد در SAT، اگر خودش بود، دانشجوی سالم و اگر نقیضش بود، دانشجوی مریض را در آن ردیف مربوط به آن clause و آن ستون می گذاریم.

(البته الان که فکر می کنم می توانستیم 3SAT رو هم به اون کاهش بدیم ~~[illegible]~~ (برای بزرگترین clause

البته اگر از متغیرهای مختلف در هر clause استفاده بشه، حداکثر nm متغیر مختلف داریم و n تا clause. پس یک جدول $n \times nm$ از Bola خواهیم داشت که باز هم از اردر چند جمله ای نسبت به ورودی خواهد بود. (پس یک شماره گذاری اولیه لازمه که متغیرهای مختلف رو بگیم که این شماره ها نشانه ستون آنها خواهد بود)

سوال 5) اول ثابت می‌کنیم که مسئله NP است.

چون می‌توانیم certificate مان را همان زیرگراف $[\frac{m}{2}]$ راسی در نظر بگیریم که کامل باشند، یعنی ورودی مان را چک می‌کنیم که تک تک $\frac{m}{2}$ راس‌های رنگ شده به هم یال داشته باشند (باید اول مطمئن باشیم که این $\frac{m}{2}$ راس از هم متمایز و معتبر و در گراف موجود هستند) این چک کردن یا به اصطلاح verify کردن از اردر $\binom{m/2}{2}$ است (به ازای هر دو راس در $\frac{m}{2}$ راس داده شده، چک می‌کنیم به هم یال داشته باشند) پس اگر هر دو از آنها به هم یال داشته باشند، زیرگراف کامل با $\frac{m}{2}$ راس داریم و جواب بله است، در غیر این صورت جواب خیر و نادرست است پس verify کردن یک جواب در اردر چند جمله‌ای قابل حل است.

حال اثبات NP-Hard بودن: مسئله clique را به آن کاهش می‌دهیم. به این شکل که فرض کنید ورودی clique، یک G با n راس و یک k باشد که دنبال یک کلیک k تایی در این n راس هستیم. حال یک گراف جدید G' می‌سازیم که به این شکل از روی G ساخته می‌شه: اول n تا راس اضافه می‌کنیم (پس الان 2n تا راس داریم) دوم اینکه n-k تا از راس‌های جدید را به دلخواه انتخاب می‌کنیم و این n-k تا راس را به تمام n تا راس قدیم یال می‌زنیم و وصل می‌کنیم. همچنین بین خود این n-k تا راس هم تمام یال‌ها را می‌زنیم و وصل می‌کنیم. حال قبلا که دنبال یک clique به اندازه k بودیم، چون الان n-k راسی هستن که به همه یال دارن، باید دنبال یک کلیک به اندازه $k+(n-k)=n$ در گراف جدید G' با 2n راس باشیم که دقیقا همان مسئله ما است.

اگر یک کلیک k تایی در گراف G داشته باشیم. تمام آن‌ها به هم یال دارند. چون در G' هم، n-k راس را انتخاب کردیم که هم به هم یال دارن هم به همه راس‌های قدیمی (از جمله k تای قدیمی) پس این k به علاوه n-k تای جدید، یک کلیک n راسه در گراف G' با 2n راس تشکیل می‌دن پس اگه کلیک k تایی داشتیم در G کلیک n تایی در G' خواهیم داشت.

ادامه 5) حالا اگر یک کلیک n راسی در گراف G' با $2n$ راس داشته باشیم، حداکثر $n-k$ راس آن از راس های جدید انتخاب شده (چون بقیه راس های جدید، هیچ یالی به بقیه ندارن پس تو هیچ کلیکی نیستن) پس حداقل k راس از راس های قدیمی وجود دارن که کلیک هستن. این k راس با یال های بینشان در گراف G با n راس هم هستن و در نتیجه یک کلیک به اندازه k در گراف G خواهیم داشت.

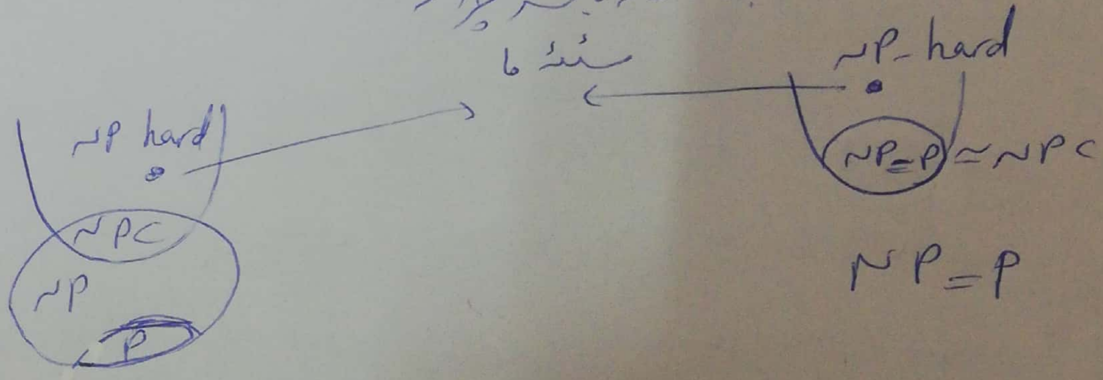
آیا کلیک به اندازه n در گراف G' با $2n$ راس داریم $\Longleftrightarrow$ آیا کلیک به اندازه k در گراف G با n راس داریم؟

ساخت گراف G' هم از اردر چندجمله‌ای است. چون n راس اضافه می‌کنیم و حداکثر هم از اردر $O(n^2)$ تا یال اضافه می‌کنیم. در کل از اردر چندجمله‌ای نسبت به ورودی راس و یال اضافه می‌کنیم تا G' ساخته بشه.

سوال 6)

الف) خیر؛ ~~نگفته که مسئله C از مسائل NP هست یا نه~~، علاوه بر اول، نگفته که C یک مسئله NPC هست یا نه، فقط گفته که به A در زمان چندجمله‌ای کاهش پیدا می‌کنه. اگر C از مسائل NPC بود یعنی در زمان چندجمله‌ای همه NP ها به آن قابل کاهش بودن، در این حالت اگر C در چندجمله‌ای حل می‌شد، تمام مسئله‌های NP هم با کاهش به آن در چندجمله‌ای، در چندجمله‌ای حل می‌شدن و آنگاه P = NP بود. درصورتی که نگفته چنین چیزی.

ب) خیر؛ ما فقط توانستیم که A را به B کاهش دهیم که این یعنی B یک مسئله NP-Hard است. اما ثابت نشده است که NP هست یا نه، ممکنه اصلا NP نباشه. مثلا فرض کنید که B یک مسئله NP hard است و NP نیست. پس هر دو حالت P=NP و P≠NP ممکن است وجود داشته باشد چرا که



NP ≠ P

NP = P

ج) درسته؛ چون اگه در زمان چندجمله‌ای حل بشه، چون NPC ها هست، بقیه NP ها هم در زمان چندجمله‌ای به اول کاهش پیدا می‌کنن و در زمان چندجمله‌ای حل می‌شن و اینطوری P=NP می‌شه که در تناقض با اینکه P≠NP. پس فرض خلف باطل و حکم ثابت می‌شه.

د) خیر؛ ما فقط می‌دونیم که A در زمان چندجمله‌ای می‌تونه به B کاهش پیدا کنه و لزومی نداره از n^2 باشه. چه‌بسا که حتی این کاهش از n^k باشه ($k \geq 1$) و B هم با n^2 حل بشه یعنی A را $O(n^{k+2})$ حل می‌شه که از $O(n^2)$ اکیداً بزرگتره. پس چندجمله‌ای هست اما لزومی نداره $O(n^2)$ باشه.