Punto 1: Recuerde que si L_1 y L_2 son 2 conjuntos de cadenas, entonces:

$$L_1 \circ L_2 = \{w_1 \circ w_2 : w_1 \in L_1 \wedge w_2 \in L_2\}$$

Demuestre que si L_1 y L_2 están en NP entonces $L_1 \circ L_2$ están en NP.

Solución:

Hecho de clase 1: Para un lenguaje $L \subseteq A^*$, las 2 siguientes afirmaciones son equivalentes
1)Existe una máquina de Turing determinista M tal que M corre en tiempo polinomial y existe un polinomio p(n) tales que:
Para toda cadena $x \in A^*$.
$x \in L \longleftrightarrow$ existe una $w \in A^*$: M acepta $< x, w >$.
2) $L \in NP$.
Podemos pensar en la cadena w como un certificado de que $x \in L$ y la máquina de Turing M la llamaremos certificador.
Note además que el certificado tiene una longitud acotada por un polinomio p(|x|) evaluado en el número de símbolos de x.

Hecho de clase 2: Verificar si un input corresponde al código para una sucesión de cadenas o corresponde al código para un grafo requiere tiempo polinomial.

Suponga que L_1 y L_2 están en NP y además tienen alfabeto A, la siguiente máquina de Turing determinista M en input y hace lo siguiente:
1) Verifica que $y =< x, m >$, si no rechaza.
2)Verifica que $x =< w_1, w_2 >$ para 2 cadenas en A^*, si no rechaza.
3) Verifica que $m =< z_1, z_2 >$ para 2 cadenas en A^*, si no rechaza.
4)Verifica que $< w_1, z_1 >\in L(N)$ y $< w_2, z_2 >\in L(D)$ donde N y D son las máquinas de Turing verificadoras para L_1 y L_2 que nos asegura el hecho de clase 1,si se cumple lo anterior acepta si no rechaza.

Teorema 1: Si L_1 y L_2 están en NP entonces $L = L_1 \circ L_2$ está en NP .
Demostración:
Empecemos demostrando que la máquina de Turing M es un certificador.
Veamos que el número de pasos para la máquina de Turing M es polinomial.
Paso 1,2,3: Por el hecho de clase 2 estos pasos requieren tiempo polinomial.
Paso 4: verificar que $< w_1, z_1 >\in L(N)$ y $< w_2, z_2 >\in L(D)$ requiere tiempo polinomial ya que M simula a N en input $< w_1, z_1 >$ y después simula a D en input$< w_2, z_2 >$ y ambas simulaciones requieren un número de pasos polinomial por el hecho de clase 1.
Trivialmente notamos que el tiempo es polinomial para esta máquina de Turing, adicionalmente podemos acotar la longitud del certificado por el polinomio p(|x|)=$|< z_1, z_2 >|$=$p_1(|x|) + p_2(|x|) + 1$, donde p_1 y p_2 son los polinomios que acotan los certificados para N y D.
El lenguaje de esta máquina de Turing es $\{< w_1, w_2 >: w_1 \in L_1 \wedge w_2 \in L_2\}$, el anterior lenguaje es equivalente a $L_1 \circ L_2$(se deja al lector comprobarlo) y por el hecho de clase 1 $L_1 \circ L_2$ está en NP.
Q.E.D

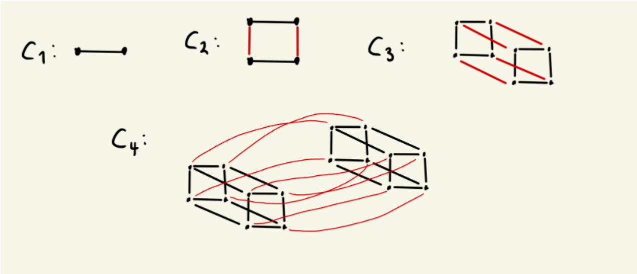

Figure 1:

Punto 2: Definimos,por recursión, los grafos (no dirigidos) $C_1, C_2, C_3, ...$ así: C_1 es el grafo con 2 vértices conectados por una arista, y C_{n+1} se toma tomando 2 copias de C_n y conectando con una arista cada vértice de la primera copia de C_n a su "gemelo" en la segunda copia de C_n. Abajo se muestran C_1, C_2, C_3 y C_4.($Figure$ 1)
a) Demuestre que para todo $n \in 1, 2, ...$, el grafo C_n tiene un camino hamiltoniano.
b) Demuestre que para todo $n \in 1, 2, ...,$,

$$< C_n, 2^{n-1} >\in VERTEX - COVER$$

Como definido en la sección 7.5 de Sipser: es decir,existe un conjunto X de 2^{n-1} vértices de C_n tal que cada arista de C_n toque un vértice en X.

Solución:

Definición 1: Un camino Hamiltoniano: es un camino simple (que no repite vértices) que incluye todos los vértices de G.

a)**Demostración:**
Por inducción sobre n(enésimo grafo de la lista definida).
Caso base n=1:
Trivialmente C_1 tiene un camino hamiltoniano..
Hipótesis de Inducción:
H(n): Para el grafo C_n,C_n tiene un camino Hamiltoniano.
Paso inductivo:
Suponga H(n), construya C_{n+1} y elija el siguiente camino:
Empiece por alguna copia de C_n de C_{n+1} en esa copia siga el camino hamiltoniano que asegura la hipótesis de inducción, cuando finalice el camino hamiltoniano, pase al vértice "gemelo" y recorra el camino hamiltoniano anterior en orden contrario a través de la otra copia de C_n en C_{n+1}.
Evidentemente se han recorrido los vértices una sola vez, por ende
$H(n) \longrightarrow H(n+1)$.
$Q.E.D$
b)**Demostración:**
Por inducción sobre n(enésimo grafo de la lista definida).
Caso base n=1:
Trivialmente C_1 tiene solo un vértice tal que cada arista toca a este mismo, es decir $< C_1, 2^{1-1} >=< C_1, 2^0 >=< C_1, 1 >\in VERTEX - COVER$.

Hipótesis de Inducción:
H(n): Para el grafo C_n $< C_n, 2^{n-1} >\in VERTEX - COVER$
Paso inductivo:
Utilicemos el hecho de que cada grafo tiene 2^n vértices(El lector pruebe comprobar este hecho por inducción, la idea intuitiva es que empezamos con 2 vértices y en cada paso se van duplicando el número de vértices).
Suponga H(n),a través de C_n construya a C_{n+1} y siga el siguiente algoritmo:
1) Para alguna copia de C_n en C_{n+1} construya el conjunto X de vértices que asegura la hipótesis de inducción tal que $< C_n, X >\in VERTEX - COVER$ y $|X| = 2^{n-1}$.
2) Utilizando el hecho mencionado al principio, notamos que el conjunto X de vértices es la mitad del total de vértices que están en la primera copia C_n de C_{n+1}. Añada al conjunto X el conjunto de vértices Y en la otra copia de C_n en C_{n+1} tal que ningún elemento w en Y sea "gemelo" de algún elemento de X.
Note que el conjunto Y de elementos añadidos w cumple que para la otra copia de C_n en C_{n+1} $< C_n, Y >\in VERTEX - COVER$, podemos justificar esto de la siguiente manera:
• Y tiene 2^{n-1} elementos ya que X tiene 2^{n-1} elementos y la cantidad de elementos de la otra copia de C_n en C_{n+1} tiene 2^n elementos por el hecho mencionado al principio, así quedaría que la cantidad de "gemelos" de la segunda copia("gemelos que no conectan con ningún elemento de X") es 2^{n-1}.
• Si marcamos el conjunto X de vértices de la primera copia de C_n en C_{n+1} con un símbolo especial, digamos ** y marcamos el conjunto Y de la otra copia de C_n en C_{n+1} también con el anterior símbolo; se puede pensar que al ver la segunda copia de C_n en C_{n+1} simplemente estamos viendo la primera copia rotada, o mirando la otra cara de la primera copia, de igual manera podríamos pensar que si tenemos a una copia de C_n(con los puntos X y Y en ella)los puntos X captan un vértice para cada arista y los puntos Y captan la "el otro vértice para las aristas",por ende el conjunto Y cumple con lo que queríamos justificar. Para acortar la solución, se deja al lector la comprobación de este hecho(esto se puede hacer con los anteriores argumentos).
Tenemos por lo tanto que para cada arista en la primera copia C_n en C_{n+1} un vértice de estas misma toca un elemento del conjunto $X \cup Y$ y análogamente que para cada arista en la segunda copia C_n en C_{n+1} un vértice de esta misma toca un elemento del conjunto $X \cup Y$. Por último note que el conjunto de aristas Z que conectan cada "gemelo" de las 2 copias tienen un vértice en el conjunto $X \cup Y$, ya que la mitad de los elementos de Z tienen un vértice en X y por la construcción anterior de Y la otra mitad disyunta de elementos en Z tienen un vértice en Y. Todas las aristas posibles de C_{n+1} tienen un vértice en $X \cup Y$. En conclusión $< C_{n+1}, X \cup Y >\in VERTEX - COVER$ y además al ser X y Y disyuntos $|X \cup Y| = |X| + |Y| = 2^{n-1} + 2^{n-1} = 2^n$, se cumple entonces H(n+1) y tenemos que H(n)$\longrightarrow$H(n+1).
Q.E.D

Punto 3: Muestre que:
$CONNECTED = \{< G >$: es un grafo no dirigido y conexo $\}$
está en P

Solución:

Nota 1: La respuesta está basada en el ejemplo 3.23(pág 185 y

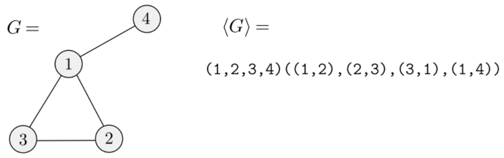

Figure 2:

186)*(Introduction to the Theory of Computation, 3a edición, por Michael Sipser)*. Sin embargo solo se se presenta el algoritmo, en esta solución demostraré que el algoritmo está en P. Para recordar el código para un grafo vea la imagen (*Figure 2*).
Definición 2: Un grafo conexo es aquel en el que cada par de vértices está conectado por un camino.

Sea $A = \{< G >=$ G es un grafo conexo y no dirigido$\}$
La siguiente máquina de Turing determinista M decide a A, en input x:
1) Verifica si $x =< G >$ para algún grafo G no dirigido.
2) Selecciona el primer vértice de G y lo marca.
3) Repite el siguiente proceso hasta que no existan nuevos vértices para ser marcados:
Para cada vértice en $< G >$, márquelo si está unido por una arista a un vértice que ya está marcado.
4)Escanee todos los vértices de G para determinar si todos están marcado. Si lo están, acepte; de lo contrario, rechace.
La solución funciona por lo siguiente:
En el primer paso se verifica que el input sea el código para un grafo G no dirigido, después en el segundo paso selecciona el primer vértice de G y lo marca, después fijémonos bien en el tercer paso, marca primero los vértices que conectan directamente con el primer vértice de G, después si sobran vértices marca los que conectan "a través de 2 pasos" con el primer vértice o 1 paso con los últimos vértices recién marcados; así se hace sucesivamente hasta que se lleguen a los vértices que conectan "a través de n pasos" con el primer vértice o "a través de n-1 paso o menos" con los anteriores vértices marcados, a través de este proceso es evidente que 2 vértices que estén marcados tienen un camino o están conectado(El lector puede comprobarlo con varios ejemplos si desea) . En el paso 4 verifica si todos los vértices están marcados, es decir si todos los vértices conectan "a través de n pasos o menos " o equivalentemente si el grafo es conexo(ver definición 2), de lo contrario rechaza.

Hecho de clase 3: Marcar un símbolo, buscar un símbolo o verificar un simbolo requiere un número de pasos polinomial en la longitud del input.
Teorema 2: El problema de decidir si un grafo no dirigido es conexo está en P.
Demostración :
Demostremos que la anterior máquina de Turing determinista M que decide si un grafo es conexo o no corre en tiempo polinomial, esto es equivalente a demostrar de que el problema de si un grafo es conexo o no está en P.
Paso 1:Por el hecho de clase 2 esto requiere un número de pasos polinomial en

la longitud del input.
Paso 2: Por el hecho de clase 3 este paso requiere un número de pasos polinomial en la longitud del input.
Paso 3: Para cada vértice k marcado(empezamos con el primer vértice marcado (ver paso 2)) buscamos los vértices w que conecten con este vértice k a través de una arista y marcamos los vértices w(Esto requiere un número de pasos polinomial en la longitud del input por el hecho de clase 3).Sin pérdida de generalidad suponga que para cada vértice k buscar los vértices w que conecten con él a través de una arista y marcarlos tiene complejidad temporal $O(n^d)$(Siendo n la longitud del input).
Para cada vértice realizamos el proceso anterior, por ende el número de pasos es O(número vértices*n^d)=$O(n^{d+1})$.
Paso 4: Verifica que cada vértice esté marcado (Esto requiere un número de pasos polinomial en la longitud del input para cada vértice por el hecho de clase 3 y análogamente al paso 3 tenemos que al realizar el proceso sobre cada vértice el número de pasos sigue siendo polinomial).
La cantidad de pasos necesarios entonces es polinomial para la máquina de Turing M determinista que decide si un grafo no dirigido es conexo, por ende este problema está en P.
Q.E.D

Punto 4: En la clase, definimos:
$HAMILTON$=$\{< G >$: es un grafo no dirigido con un circuito hamiltoniano $\}$
En el libro definimos a :
$UHAMPATH$=$\{< G, s, t >$: G es un grafo no dirigido y existe un camino hamiltoniano desde el vértice s hasta el vértice t$\}$(Un camino "hamiltoniano" es un camino desde s hasta t que pasa una sola vez por cada vértice del grafo).
Demuestre que $HAMILTON \leq_P UHAMPTH$ y que
$UHAMPT \leq_P HAMILTON$

Solución:

Nota: La solución fue explicada a mi por mi compañero Diego Ramírez, sin embargo aquí presento la solución detalladamente y a mi manera; ya que entendí perfectamente como se soluciona este ejercicio.
Definición 3: Una función $f : \Sigma^* \to \Sigma^*$ es calculable en tiempo polinomial si existe una máquina de Turing M y un $k \in \mathbb{N}$ tales que en cualquier input con $|w| = n$, la máquina M se para en $O(n^k)$ pasos con $f(w)$ en su cinta.
Definición 4: Si A y B son 2 lenguajes sobre el alfabeto Σ, decimos que A es reducible a B por un mapa polinomial si existe alguna función $f : \Sigma^* \to \Sigma^*$ que es calculable en tiempo polinomial, además se cumple que para toda cadena w, $w \in A \longleftrightarrow f(w) \in B$.Denotamos que A es reducible por un mapa polinomial a B por $A \leq_P B$.
Definición 5: Un Circuito Hamiltoniano es un camino cerrado que pasa una sola vez por todos y cada uno de los vértices del grafo, es decir, es un ciclo que a su vez es un camino hamiltoniano.
Para $HAMILTON \leq_P UHAMPTH$ utilice el siguiente mapeo polinomial:
• Como un circuito hamiltoniano empieza y termina en el mismo sentido, podemos escoger un vértice arbitrario k, como veremos este vértice tendrá mínimo 2 aristas conectadas a él(Ver lema 2). Creamos un nuevo vértice k' arbitrario tal que k' conecte con todos los vértices que conectan directamente

con k, retorne por último $< R, k, k' >$, donde $< R >$ es el grafo resultante. Evidentemente este mapeo se puede realizar en pasos polinomiales.(Se deja al lector comprobarlo para acortar la solución.)
Teorema 3: $< G >\in HAMILTON \longleftrightarrow < R, k, k' >\in UHAMPTH$ y $UHAMPTH$es NP-completo.
Demostración:
" $< G >\in HAMILTON \longrightarrow < R, k, k' >\in UHAMPATH$".
Suponga que $< G >\in HAMILTON$ entonces evidentemente en el grafo $< R >$ existe un camino hamiltoniano desde k hasta k' ya que en vez de finalizar el circuito en k,se termina en k' para generar un camino hamiltoniano desde k hasta k'(Note que todos los vértices se recorrieron solo una vez), por ende $< R, k, k' >\in UHAMPATH$.
" $< R, k, k' >\in UHAMPATH \longrightarrow < G >\in HAMILTON$".
Por contra reciproca:
Suponga que $< G >\notin HAMILTON$ por ende no existe un circuito hamiltoniano tal que este mismo se pueda cortar justo antes de conectar con k de nuevo y "encaminarse" a k' recorriendo así todos los vértices y generando un camino hamiltoniano desde k a k', por tal motivo
$< G, k, k' >\notin UHAMPATH$.
Por el hecho de clase 5 hemos reducido un problema NP completo a $UHAMPATH$, por ende $UHAMPATH$ es NP-completo(Necesitaremos el hecho de que este problema es NP-completo después).
$Q.E.D$
Para $UHAMPATH \leq_P HAMILTON$ utilice el siguiente mapeo polinomial:
• Dado un código $< G, s, t >$ tal que $< G >$ es un grafo y s y t son 2 vértices de este mismo, añada el vértice v y añada las aristas vs y vt,después retorne el grafo $< R >$ resultante. Evidentemente este mapeo se puede realizar en pasos polinomiales.(Se deja al lector comprobarlo para acortar la solución.)
Teorema 4: $< G, s, t >\in UHAMPATH \longleftrightarrow < R >\in HAMILTON$.
Demostración:
" $< G, s, t >\in UHAMPATH \longrightarrow < R >\in HAMILTON$".
Suponga que $< G, s, t >\in UHAMPATH$ entonces evidentemente en el grafo $< R >$ existe un circuito hamiltoniano que empieza y termina en s ya que en vez de finalizar el camino hamiltoniano en t, se pasa a v y de v se pasa s.(Note que todos los vértices se recorrieron solo una vez y se termino e inició en el mismo punto), por ende $< G, s, t >\in UHAMPATH$.
" $< R >\in HAMILTON \longrightarrow < G, s, t >\in UHAMPATH$".
Por contra reciproca:
Suponga que $< G, s, t >\notin UHAMPATH$ Entonces hay varias posibilidades, 2 de ellas son:
$*$ Existe un camino simple entre s y t pero ese camino no recorre a todos los vértices. Cuando se construye $< R >$ notamos que si ya existía un circuito hamiltoniano $< G, s, t >$ entonces las aristas agregadas "arruinan ese camino", de igual manera se nota que si no existía un camino hamiltoniano, el hecho de agregar las aristas "no cambia nada". Por ende no podemos realizar un circuito hamiltoniano.
$*$ No existe camino entre s y t. En este caso el grafo no es conexo, por lo tanto "habría mínimo 2 mitades conexas", sin embargo aunque se añada el vértice v que conecte a s y a t no se podría construir un circuito hamiltoniano ya que se puede pasar de una parte conexa a otra parte conexa a través de las aristas

creadas, pero aún así para regresar habría que pasar de nuevo por las aristas creadas, repitiendo vértice Por ende no podemos realizar circuito hamiltoniano. Para acortar la solución se deja al lector los otros caso y la comprobación formal de cada uno de ellos.
Q.E.D

Punto 5: Sea $3HAMILTON$ el conjunto de todos los códigos $< G >$ tales que $< G >$ es un grafo no dirigido, cada vértice de G tiene grado por lo menos 3, y G tiene un circuito hamiltoniano. Demuestre que $3HAMILTON$ es NP-completo.

Solución:

Hecho de clase 5: Si un problema X que es NP completo es reducible por un mapa polinomial a otro problema Y que está en NP, entonces Y es NP-completo.
Dado a que $3HAMILTON$ está en NP(Ver el libro guia),basta demostrar que algún problema NP completo puede reducirse mediante un mapa polinomial a $3HAMILTON$ para afirmar que $3HAMILTON$ es NP-completo.
Lema 1: Que un grafo sea conexo es una condición necesaria, pero no suficiente para que el grafo tenga un circuito hamiltoniano.
Demostración:
Por contradicción:
•Suponga que existe un grafo que tiene un circuito hamiltoniano y que no sea conexo. Que no sea conexo, por definición asegura que existe por lo menos un par de vértices a y b que no están conectados por un camino, pero el grafo tiene un circuito hamiltoniano y por definición esto asegura que existe circuito que pasa por cada uno de los vértices incluyendo el par de vértices que no están conectados, es decir existe un camino entre a y b**(Contradicción)**.
• Contra-ejemplo de grafo conexo pero sin circuito hamiltoniano.(*figure 3*)
Q.E.D
Lema 2: Una condición necesario pero no suficiente para que un grafo tenga un circuito hamiltoniano es que cada vértice tenga por lo menos grado 2 (es decir tenga al menos 2 vértices conectados a él por una arista).
Demostración:
Por contradicción:
Suponga que existe un grafo con circuito hamiltoniano tal que en un vértice este conecta con 1 o con ningún otro vértice a través de una arista.
• Si existe un vértice k que no conecta con ninguno otro entonces el grafo es disconexo, pero hay un circuito hamiltoniano, pero esto contraria el lema 1 **Contradicción**.
• Si existe un vértice k que conecta solo con otro vértice w, entonces empiece el circuito hamiltoniano asegurado desde ese vértice k, vaya al otro vértice w a través de la arista, por hipótesis existe un circuito hamiltoniano por ende se debe volver al punto inicial k pasando una sola vez por cada vértice, sin embargo la única forma de volver al punto inicial k es "pasando" de w a k; pero ya recorrimos una sola vez a w y a k, es decir el circuito hamiltoniano tendría que pasar 2 veces por w **Contradicción**.
• Contra-ejemplo de grafo con cada vértice con grado 2 o más que no tiene un circuito hamiltoniano.(*figure 3*).*Note además que 2 condiciones necesarias pero no suficientes para que un grafo contenga un circuito hamiltoniano es que*

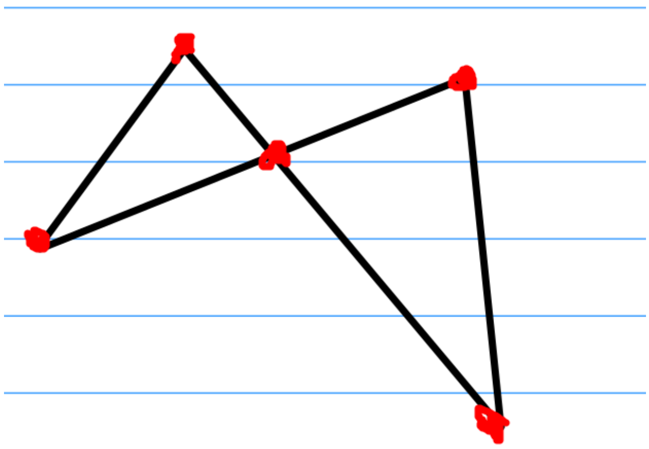

Figure 3:

sean conexos y cada vértice sea de grado por lo menos 2, vea la figura 3 de nuevo para que corrobore que son condiciones necesarias pero no suficientes.
Q.E.D
Lema 3: En grafo $< G >$ que contenga un circuito hamiltoniano cada vértice del circuito hamiltoniano conecta con 2 y solamente 2 aristas del circuito hamiltoniano(Note que el circuito hamiltoniano es un subconjunto de aristas del grafo$< G >$).
Demostración:
Por contradicción:
Suponga que un grafo $< G >$ tiene un circuito hamiltoniano y no conecta con 2 aristas del circuito hamiltoniano, tenemos 2 casos para comprobar:
• Supongamos que para un vértice k en $< G >$ el número de aristas del circuito hamiltoniano que conectan con k es menor o igual a 2, si es 0 es decir no conectan con ninguna arista entonces el circuito hamiltoniano no pasara por un vértice del grafo, si se tiene una arista del circuito hamiltoniano conectada a k, tal que esta arista conecta a k con m, entonces el circuito podría empezar desde k a m, pero no devolverse si no solamente por la misma ruta; en ambos casos **contradicción**.
• Suponga ahora que para un vértice k en $< G >$ se tiene un número de aristas mayor o igual a 2, si recorremos el circuito hamiltoniano empezando por el vértice k y eligiendo una arista arbitraria de las m aristas posibles($m \geq 2$) podríamos terminar en la arista original k a través de alguna de las restantes aristas posibles; sin embargo al hacer esto no estamos conectando con algún vértice del circuito hamiltoniano y por ende no hemos pasado un vértice **Contradicción**.
Q.E.D
Corolario 1: Si $< G >$ es un grafo con circuito hamiltoniano tal que existe un vértice k con grado 2, las 2 aristas que conectan con este vértice hacen parte del subconjunto de aristas que forman el circuito hamiltoniano.
Demostración:
Suponga que $< G >$ es un grafo con circuito hamiltoniano, suponga además que existe un vértice k con grado 2. Por el lema 3 este vértice k conecta con 2

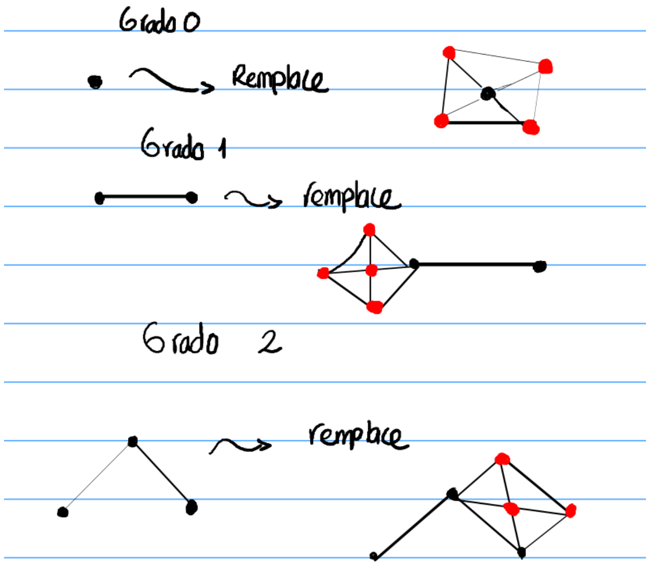

Figure 4:

aristas del circuito hamiltoniano, sin embargo solo contiene 2 aristas conectadas a él, por ende las 2 aristas que conectan con él hacen parte del circuito hamiltoniano.
Q.E.D
Lema 4: Suponga que $< G >$ es un grafo entonces existe un proceso tal que se genera un grafo con solo vértices de grado mayor o igual a 3 y el nuevo grafo tiene un circuito hamiltoniano si y solo si $< G >$ tiene un camino hamiltoniano.
Esbozo de demostración:
Siga el siguiente proceso(*figure 4*):
La idea del porque funciona por lo siguiente:
Si el grafo tiene al menos un vértice de grado 0 o 1 entonces no tiene un camino hamiltoniano por los lemas anteriores. En ambos casos solo se remplazan por un conjunto nuevo de aristas y vértices tal que tanto los nuevos vértices y los "antiguos" tengan 3 vértices conectados a él través de una arista,sin embargo para el "nuevo" grafo construido no se altera el hecho de que no tenga camino hamiltoniano.
Si el grafo tiene grado 2 entonces puede llegar de un vértice de la "arista original" al otro vértice de la "arista original" tocando solamente una vez los vértices rojos y llegando al otro extremo; por lo tanto si hay un circuito hamiltoniano sería solo cambiar levemente la ruta; y si no hay circuito hamiltoniano en el grafo original, no sigue existiendo un circuito hamiltoniano en el grafo "nuevo". Para acortar la solución se deja al lector los detalles formales de la comprobación de que la construcción funciona.Note que los "vértices añadidos" solo conectan entre ellos mismos y el vértice original que se remplazó.
Note además que se puede comprobar que la construcción anterior se puede realizar por una función f calculable en pasos polinomiales

Para realizar el la reducción por un mapa polinomial de un grafo $< G >$ a un grafo $< R >$ con solo vértices de grado mayor o igual a 3 utilizamos la función f anterior. Note que por el lema 4
$< G >\in HAMILTON \longleftrightarrow < R >\in 3HAMILTON$. Por lo tanto
$HAMILTON <_P 3HAMILTON$, pero sabemos que $HAMILTON$ es NP completo por ende lo anterior es equivalente a que $3HAMILTON$ es NP completo(Ver el hecho de clase 5).
Q.E.D

Punto 6: Sea $< G >$ la representación de un grafo no dirigido. Tenemos:
$SPATH) = \{< G, a, b, k >:< G >$ contiene un camino simple de longitud de máximo k pasos desde a hasta b$\}$.
$LPATH) = \{< G, a, b, k >:< G >$ contiene un camino simple de longitud de k o más pasos desde a hasta b$\}$. Mueste que $SPATH$ está en P y $LPATH$ es NP completo.

Solución:

Definición 6: Un camino simple es un camino sin vértices repetidos, salvo quizás el primero y el último.
Para demostrar que $SPATH \in P$ definimos una máquina de Turing determinista que corra en tiempo polinomial y decida a $SPATH$
La siguiente máquina de Turing determinista decide a $SPATH$ en input x:
1) Verifica que x tiene la forma $x =< G, a, b, k >$.
2) Busca el vértice a.
3) Realiza el mismo proceso que realiza la máquina de Turing que decide a $CONNECTED$ solamente que en este proceso se busca el vértice b y solo se realiza el proceso a lo sumo k veces desde el vértice a, además acepta si se encuentra a b en k pasos o menos acepta el input, si no se encuentra en a lo sumo k pasos, rechaza.
La máquina anterior corre en tiempo polinomial, la justificación es análoga a la máquina de Turing del punto 3.
Tenemos una máquina de Turing determinista que decide $SPATH$ en tiempo polinomial por ende $SPATH \in P$.
Q.E.D
Para la otra parte del punto demostremos primero que existe una máquina de Turing determinista que certifica una solución al problema en tiempo polinomial.
La siguiente máquina de Turing es una verificadora para $LPATH$
1) Verifica que x tiene la forma $x =< G, a, b, k, c >$ donde c es un camino.
2) Chequea que para el camino c ningún vértice se repita.
3) Chequea que la longitud del camino de c sea mayor o igual a k.
4) Si c satisface las condiciones anteriores acepta, de otra manera rechaza.
Note que todos los pasos se pueden hacer en tiempo polinomial(Se deja al lector comprobarlo para acortar la solución) adicionalmente el certificado se puede acotar polinomialmente por el número n de vértices del grafo, ya que ningún camino tiene longitud mayor que n .
Tenemos una máquina de turing M certificadora por lo tanto $LPATH \in NP$.
Hagamos una reducción de $UHAMPATH \leq_P LPATH$, como $UHAMPATH$ es NP completo lo anterior es equivalente a mostrar que $LPATH$ es NP completo.

Considere un código $< G, a, b >$ de $UHAMPATH$ problem, donde $< G >$ es un grafo.
El mapeo convierte al código para el grafo en $< G, a, b, k >$ donde k es el número de vértices de $< G >$. De nuevo para acortar la solución se deja al lector comprobar que los pasos se pueden realizar en tiempo polinomial.
Teorema 5: $< G, a, b > \in UHAMPATH \longleftrightarrow < G, a, b, k > \in LPATH$ y LPATH es NP-completo.
Demostración:
"$< G, a, b > \in UHAMPATH \longrightarrow < G, a, b, k > \in LPATH$".
Suponga que $< G, a, b > \in UHAMPATH$ entonces evidentemente el camino hamiltoniano desde a hasta b tiene un número de pasos igual a k y pasa solamente una vez por cada vértice, por ende $< G, a, b, k > \in LPATH$.
"$< G, a, b, k > \in LPATH \longrightarrow < G, a, b > \in UHAMPATH$".
Por contra reciproca:
Suponga que $< G, a, b > \notin UHAMPATH$ por ende $< G, a, b, k > \notin LPATH$ ya que ni siquiera existe un camino hamiltoniano que es el camino simple con "máxima longitud" posible.
Por el hecho de clase 5 (ya que reducimos $UHAMPATH$ a $LPATH$) $LPATH$ es NP-completo.
$Q.E.D$