"Año del Fortalecimiento de la Soberanía Nacional "

Universidad Nacional de Ingeniería

Trabajo de investigación

Facultad de Ciencias

El problema del horario

Integrantes:

*** Irán Patrick Adrian Chavez Macedo, 20210079G**

*** Sergio Diego Huanca Figueroa, 20202172A**

*** Franklin Espinoza Pari 20210135D**

*** Joel Jhotan Chavez Chico 20210058J**

*** José Carlos Maldonado Jesusi 20210016E**

Profesor:

RONALD MAS HUAMAN

Julio 2022

Índice

1. Resumen

Nuestro proyecto busca mostrar una solución ante el gran problema que supone la generación de horarios, para ello nos basamos en la gran teoria de grafos especialmente en la coloración, además cabe precisar que nuestro algoritmo siempre buscara la minima cantidad de periodos en un horario que genere asi cabe precisar de que los horarios que se generan no son unicos.

1.1. Introducción

El problema de encontrar un horario con ciertas condiciones impuestas primero por uno, siempre ha sido un problema, en este proyecto buscamos mostrar una solución con ciertas restricciones y con la ayuda de la teoría de grafos.

1.2. Objetivos

1. Lograr la contrucción de un horario.

2. Determinar un algoritmo que nos permita colorear grafos con su número cromatico

3. Mostrar una aplicación la coloración de grafos

2. Fundamento Teórico

2.1. Conceptos Previos

Para resolver este problema, es necesario explicar unas cosas primero:

2.1.1. Grafo:

Un grafo es un conjunto de elementos llamados vértices unidos por curvas o líneas llamadas aristas que permiten representar relaciones binarias entre elementos de un conjunto.

- Grafo no dirigido
  Es un tipo de grafo en el cual las aristas representan relaciones simétricas y no tienen un sentido definido, a diferencia del grafo dirigido, en el cual las aristas tienen un sentido y por tanto no son necesariamente simétricas, consideraremos este grafo por el momento.

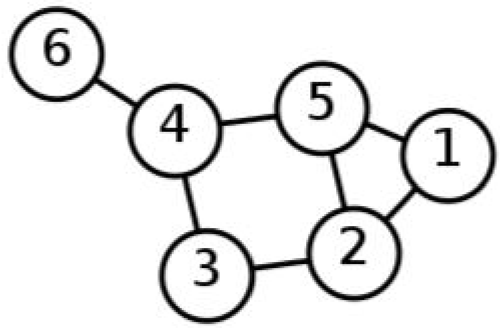


Figura 1: Ejemplo de un grafo con 6 vértices y 7 aristas

2.1.2. Grafo bipartito

Un grafo bipartito es un grafo cuyos vértices se pueden separar en dos conjuntos disjuntos, de manera que las aristas no pueden relacionar vértices de un mismo conjunto.

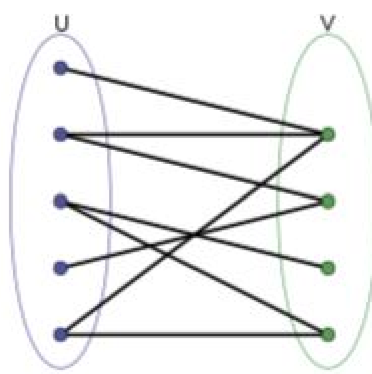


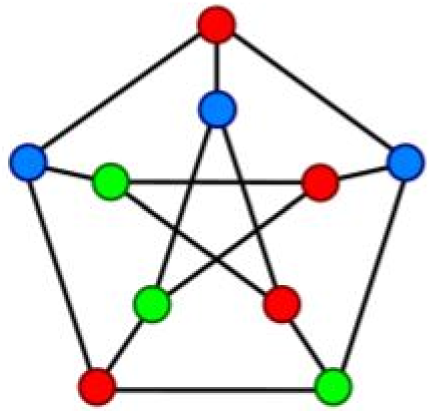
Figura 2: Ejemplo de grafo bipartito

2.1.3. Coloración de vértices

La coloración de grafos es una asignación de etiquetas llamadas colores a elementos del grafo. De manera simple, una coloración de los vértices de un grafo tal que ningún vértice adyacente comparta el mismo color es llamado: “vértice coloración”.

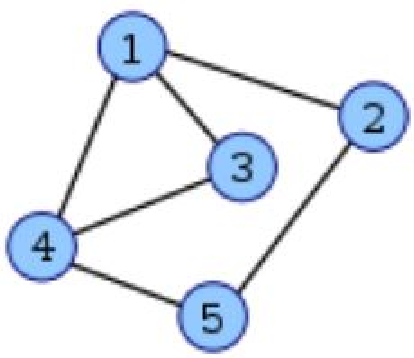
2.1.4. Grafo lineal

El grafo lineal $L(G)$ de un grafo no dirigido G es un grafo que representa las adyacencias entre las aristas de G. O sea, cumple estas dos condiciones:

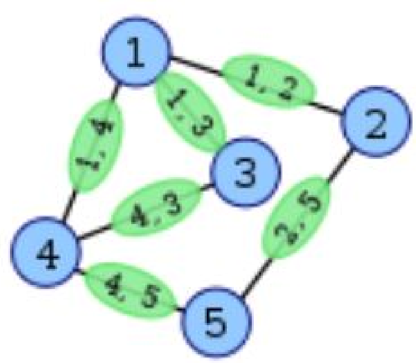


a) Cada vértice de $L(G)$ representa una arista de G.

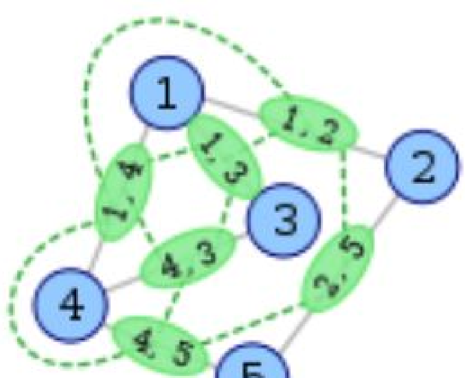
b) Dos vértices de $L(G)$ son adyacentes si y solo si sus aristas correspondientes comparten un punto en común (es decir, son adyacentes) en G.



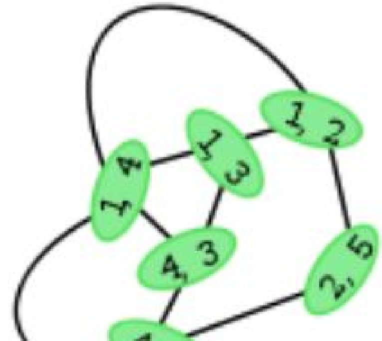
(a) Grafo G



(b) Vértices en $L(G)$



(a) Aristas añadidas en $L(G)$



(b) El grafo lineal $L(G)$

2.1.5. Algoritmo de coloración de vértices

Dado como entrada un grafo simple G con n vértices, busque una m-coloración adecuada de los vértices de G. Sea $\{u_1, u_2, \cdots, u_n\}$ denote los vértices de G y sea $\{v_1, v_2, \cdots v_m\}$ denotan los vértices de K_m. Generamos conjuntos independientes máximos en el producto cartesiano $G \times K_m$. En cada etapa, si el conjunto independiente obtenido tiene un tamaño de al menos n, pase a la parte III.

- **Parte I**

  Para $i = 1, 2, ..., n$ y $j = 1, 2, ..., m$ a su vez inicialice el conjunto independiente $S_{i,j} = \{(u_i, v_j)\}$.

  Realice el procedimiento 3.1 en $S_{i,j}$ Para $r = 1, 2, ..., n$, realice el procedimiento 3.2 repetido r veces.

  El resultado es un conjunto independiente máximo $S_{i,j}$

- **Parte 2**

  Para cada par de conjuntos independientes máximos $S_{i,j}$, $S_{k,l}$ encontrados en la Parte I

  Inicialice el conjunto independiente $S_{i,j,k,l} = S_{i,j} \cap S_{k,l}$

  Realice el procedimiento 3.1 en $S_{i,j,k,l}$

  Para $r = 1, 2, ..., n$, realice el procedimiento 3.2 repetido r veces.

  El resultado es un conjunto independiente máximo $S_{i,j,k,l}$

- **Parte III**

  Si se ha encontrado un conjunto independiente S de tamaño n en cualquier etapa de la parte I o la parte II, genere S como una coloración m adecuada de los vértices de G de acuerdo con el lema cartesiano. De lo contrario, informe que el algoritmo no pudo encontrar ninguna m-coloración adecuada de los vértices de G.

2.1.6. Algoritmo hallar el grafo lineal

Primero etiquetamos los vértices del grafo que tengamos, construiremos el grafo lineal de la siguiente forma

1. Cada arista del grafo inicial será un vértice para su grafo lineal.

2. Si las aristas en G compartían un vértice en común entonces en el grafo lineal dichos vértices serán adyacentes.

Notemos el grafo línea es el grafo de intersección de las aristas de G, nosotros en particular nos interesara hallar la matriz de adyacencia por ello de la misma definición construiremos la matriz de adyacencia, una vez etiquetados si las aristas tenían un vértice en común en el grafo G entonces en el grafo lineal $L(G)$ estarán relacionados por ello tendrán un 1 sino un 0, además notemos que el grafo lineal no es multígrafo por ello el nombre de este.

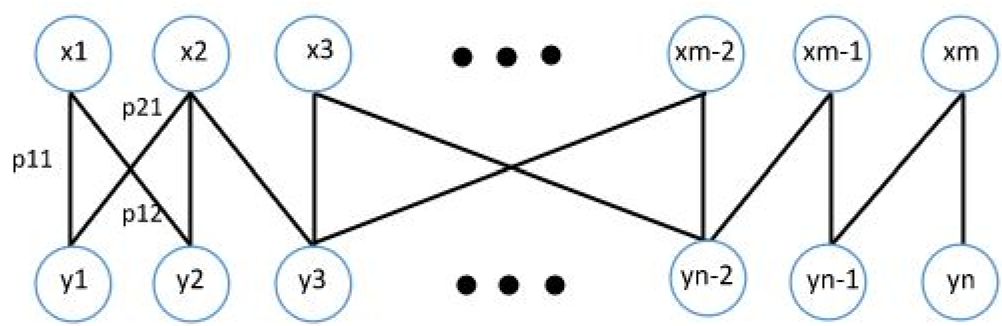
3. Planteamiento del problema

3.1. Explicacion del problema

En un colegio hay m profesores $x_1, x_2, ..., x_m$ y n asignaturas $y_1, y_2, ..., y_n$ para impartir. Dado que el profesor x_i debe (y puede) enseñar la materia y_j durante p_{ij} períodos ($p = [p_{ij}]$ se denomina matriz de requisitos de enseñanza), la administración de la universidad desea hacer un horario utilizando el mínimo número posible de periodos.

3.2. Solución teórica

Esto se conoce como el problema del horario y se puede resolver utilizando la siguiente estrategia. Construya un multígrafo bipartito G con vértices $x_1, x_2, ..., x_m,\ y_1, y_2, ..., y_n$ tal que los vértices x_i e y_j estén conectados por aristas p_{ij}.



Consideremos lo siguiente:

- En cualquier período cada profesor puede enseñar como máximo una materia.
- Cada materia puede ser impartida por un profesor como máximo.

Consideremos, primero, un solo período. El horario de este único período corresponde a una coincidencia en el gráfico y, por el contrario, cada coincidencia corresponde a una posible asignación de profesores a las materias impartidas durante este período.
Así, la solución al problema del horario consiste en dividir las aristas de G en el mínimo número de coincidencias.
De manera equivalente, debemos colorear correctamente los bordes de G con el mínimo número de colores.
Nosotros mostraremos otra forma de resolver el problema utilizando el algoritmo de coloración de vértices. Recuerde que el grafo lineal $L(G)$ de G tiene como vértices las aristas de G y dos vértices en $L(G)$ están conectados por una arista si y sólo si las aristas correspondientes en G tienen un vértice en común.
El grafo lineal $L(G)$ es un grafo simple y una coloración de vértice adecuada de $L(G)$ produce una coloración de borde adecuada de G utilizando el mismo número de colores. Por lo tanto, para resolver el problema de los horarios, basta con encontrar una coloración de vértice mínima adecuada de $L(G)$ usando el algoritmo de coloración de vértices. Demostramos la solución con un pequeño ejemplo.

3.3. Aplicación

Realizamos la aplicación a un grafo simple primero:

- Sean 5 profesores y 6 asignaturas, podemos definir la siguiente matriz de requisitos de enseñanzas:

P	y_1	y_2	y_3	y_4	y_5	y_6
x_1	1	1	0	0	0	0
x_2	0	0	0	1	1	1
x_3	0	1	0	1	0	0
x_4	1	1	1	0	0	1
x_5	0	1	0	0	0	0

Tomemos:

y_1: Calculo Diferencial

y_2: Calculo Integral

y_3: Calculo Avanzado

y_4: Matemática Discreta

y_5: Estructuras Algebraicas

y_6: lógica y teoría de Conjuntos

x_1: Mas Huamán Ronald Jesús

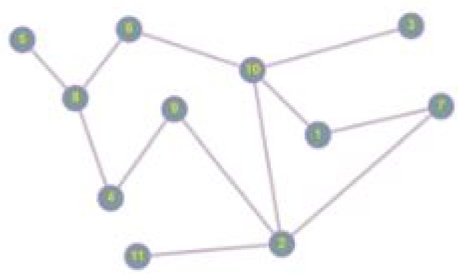
x_2: Zamudio Peves José Fernando

x_3: Acuña Ortega Richard Flavio

x_4: Metzger Alvan Roger Javier

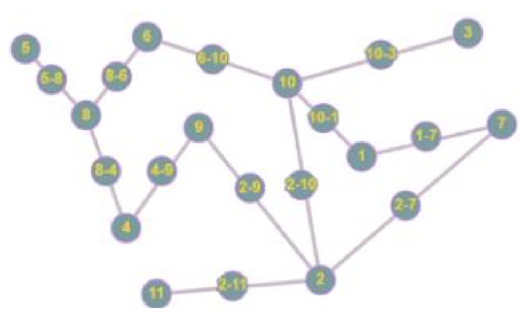
x_5: Jorge Joel Sulca Chipana

- Representemos el grafo bipartito de la siguiente forma notemos que en este caso no será multígrafo además del 1 al 6 seran los cursos y del 7 al 11 seran los profesores

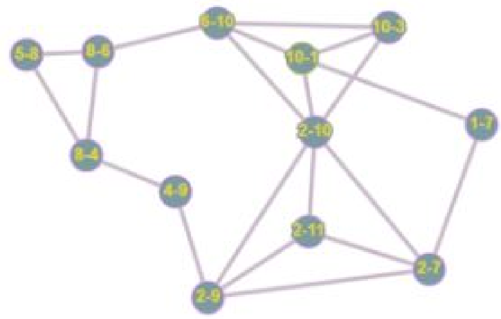


- Calculemos su grafo lineal con el algoritmo para calcular el grafo lineal.

- Asignando etiquetas a las aristas



- El grafo lineal será:



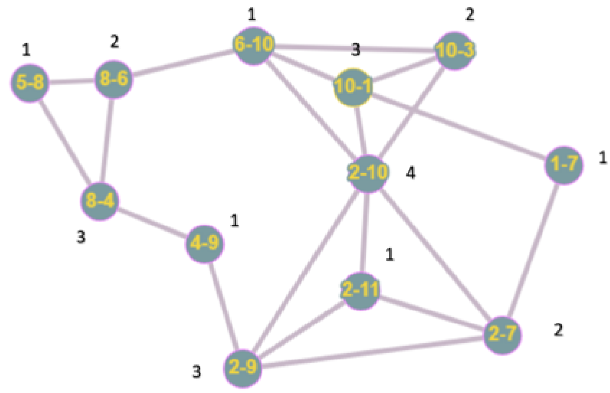
- Y su matriz de adyacencia será:

	1	2	3	4	5	6	7	8	9	10	11	12
1	0	1	0	0	0	0	0	0	0	0	0	1
2	1	0	1	0	0	0	0	0	0	0	0	1
3	0	1	0	1	1	0	1	0	0	0	0	0
4	0	0	1	0	1	0	1	0	0	0	0	0
5	0	0	1	1	0	1	1	0	0	0	0	0
6	0	0	0	0	1	0	0	1	0	0	0	0
7	0	0	1	1	1	0	0	1	1	1	0	0
8	0	0	0	0	0	1	1	0	1	1	0	0
9	0	0	0	0	0	0	1	1	0	1	0	0
10	0	0	0	0	0	0	1	1	1	0	1	0
11	0	0	0	0	0	0	0	0	0	1	0	1
12	1	1	0	0	0	0	0	0	0	0	1	0

- Usando el algoritmo de coloración obtenemos que

  Vértices coloreados (12): (1,1) (2,2) (3,1) (4,2) (5,3) (6,1) (7,4) (8,2) (9,1) (10,3) (11,1) (12,3)

  y entonces el grafo coloreado sera



- Por ultimo el horario buscado sera

	1	2	3	4
x_1	y_1	x_2		
x_2	y_5	y_6	y_4	
x_3	y_4		y_2	
x_4	y_6	y_3	y_1	y_2
x_5	y_2			

es decir

	1	2	3	4
Mas Huamán Ronald	C.Diferencial	C. Integral		
Zamudio Peves José	Estructuras Algebraicas	Lógica	Discreta	
Acuña Ortega Richard	Discreta		C. Integral	
Metzger Alvan Roger	Lógica	C. Avanzado	C.Diferencial	C. Integral
Jorge Joel Sulca	C. Integral			

3.4. Implementación

Se logro implementar los algoritmos para la generación de horarios en el lenguaje de C++ se puede descargar el codigo realizado en el siguiente enlace: `https://github.com/JoelCH04/ColoracionDeHorarios.git`

4. Conclusiones

- Se logró la construcción de un horario con las condiciones expuestas ademas de implmentarlo en un lenguaje de programación
- Se explicó un algoritmo que nos permitió colorear un grafo con su numero cromático ademas de hallar este último

5. Bibliografía

- Dharwadker, Ashay; Pirzada, Shariefuddin. Applications of Graph Theory `https://www.dharwadker.org/pirzada/applications/`
- Dharwadker, Ashay. The Vertex Coloring Algorithm `https://www.dharwadker.org/vertex_coloring/`