CSE 211: Discrete Mathematics **(Due: 04/01/21)**

Homework #3

Instructor: Dr. Zafeirakis Zafeirakopoulos
Assistant: Gizem Süngü

Problem 1: Representing Graphs (10 points)

Represent the graph in Figure 1 with an adjacency matrix. Explain your representation clearly.

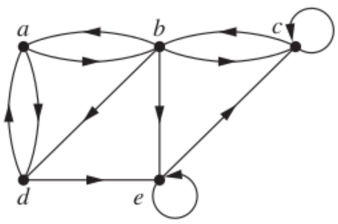


Figure 1: The graph for Problem 1

(Solution)

If a graph has n vertices, we may associate an $n \times n$ matrix M which is called vertex matrix or adjacency matrix. The vertex matrix M is defined by

$$M_{ij} = \begin{cases} 1 & \text{if } P_i \to P_j \\ 0 & \text{otherwise} \end{cases}$$

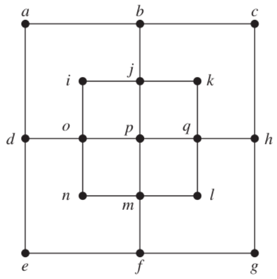
a→b=1, a→d=1
b→a=1, b→c=1, b→d=1, b→e=1,
c→b=1 c→c=1
d→a=1, d→d=1
e→c=1 e→e=1

$$\begin{array}{c c} & \begin{array}{ccccc} a & b & c & d & e \end{array} \\ \begin{array}{c} a \\ b \\ c \\ d \\ e \end{array} & \begin{pmatrix} 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix} \end{array}$$

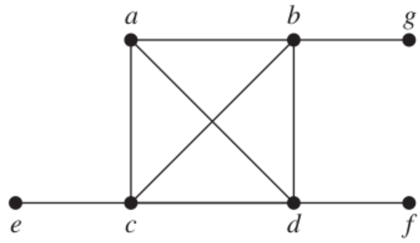
$$\begin{bmatrix} 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$

Problem 2: Hamilton Circuits (10+10+10=30 points)

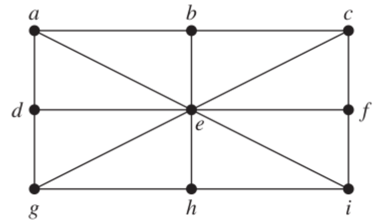
Determine whether there is a Hamilton circuit for each given graph (See Figure 5, Figure 2b, Figure 2c). If the graph has a Hamilton circuit, show the path with its vertices which gives a Hamilton circuit. If it does not, explain why no Hamilton circuit exists.



(a) The graph G_1



(b) The graph G_2



(c) The graph G_3

Figure 2: The graphs to find Hamilton circuits for Problem 1

Certain properties can be used to show that a graph has no Hamilton circuit.

- A graph with a vertex of degree one cannot have a Hamilton circuit, because in a Hamilton circuit, each vertex is incident with two edges in the circuit.
- If a vertex in the graph has degree two, then both edges that are incident with this vertex must be part of any Hamilton circuit.
- When a Hamilton circuit is being constructed and this circuit has passed through a vertex, then all remaining edges incident with this vertex, other than the two used in the circuit, can be removed from consideration. Furthermore, a Hamilton circuit cannot contain a smaller circuit within it.

(a)
(Solution)
If we construct a Hamilton circuit with vertices a-b-c-h-g-f-e-d-a, then all remaining edges incident with these vertices, other than the two used in the circuit, can be removed from consideration. So, we need to remove b-j, h-q, f-m and d-o but this process creates a smaller circuit, such as i-j-k-p-l-m-n-o-i, within our constructed Hamilton circuit. So, there is no Hamilton circuit for the graph G_1 since it violates the third property.

(b)
(Solution)
The graph has vertices of degree one: e, f and g. There is no Hamilton circuit for the graph G_2 since it violates the first property.

(c)
(Solution)
There is a Hamilton circuit for the graph G_3, such as a-b-c-f-e-i-h-g-d-a.

Problem 3: Applications on Graphs (20 points)

Schedule the final exams for Math 101, Math 243, CSE 333, CSE 346, CSE 101, CSE 102, CSE 273, and CSE 211, using the fewest number of different time slots, if there are no students who are taking:

- both Math 101 and CS 211,
- both Math 243 and CS 211,
- both CSE 346 and CSE 101,
- both CSE 346 and CSE 102,
- both Math 101 and Math 243,
- both Math 101 and CSE 333,
- both CSE 333 and CSE 346

but there are students in every other pair of courses together for this semester.
Note: Assume that you have only one classroom.

Hint 1: Solve the problem with respect to your problem session notes.
Hint 2: Check the website

(Solution)

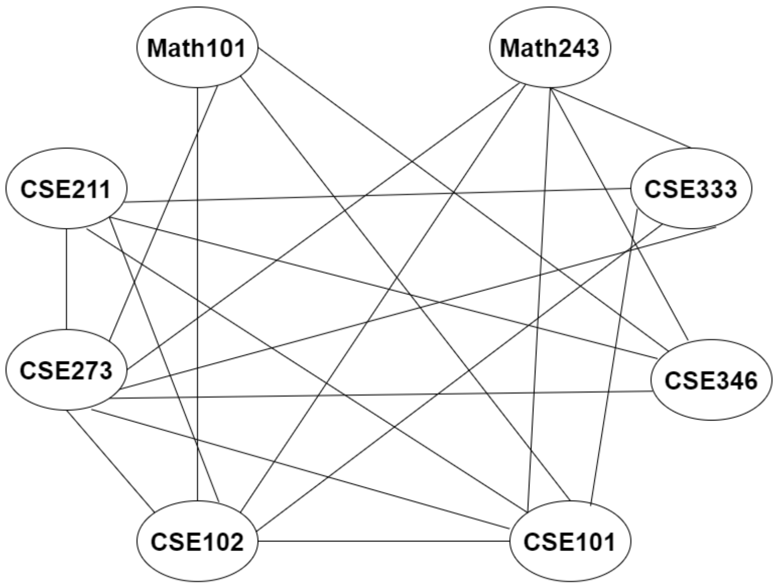


Figure 3: The graph for courses taken together

According to the graph above, courses which are not adjacent to each other, can have the same time slot. So, for all courses, we have these options for the same time slots:

Math101: Math243, CSE333 , CSE211
Math243: Math101, CSE211
CSE211: Math101, Math243
CSE333: Math101, CSE346
CSE346: CSE333, CSE101, CSE102
CSE101: CSE346
CSE102: CSE346
CSE273: Nothing

As a result, CSE273 can not share any time slot. CSE333 and CSE102 are also not sharing any time slot. (Math101,Math243,CSE211) and (CSE346,CSE101) can have the same slots.

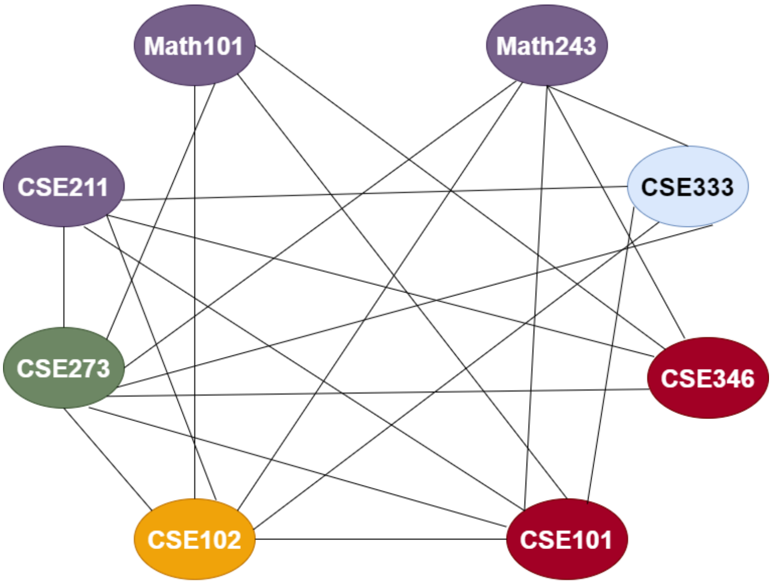


Figure 4: Graph Coloring

Time Slot / Classroom	9.00-10.00	11.00-12.00	13.00-14.00	15.00-16.00	17.00-18.00
Z23	Math101	CSE101	CSE102	CSE273	CSE333
Z23	Math243	CSE346			
Z23	CSE211				

Figure 5: Final Exams Schedule

Problem 4: Applications for Hasse Diagram of Relations (40 points)

Remember the Problem 3 in Homework 2.

Write an algorithm to draw Hasse diagram of the given relations in "input.txt" which is given for HW2.

Your code should meet the following requirements, standards and accomplish the given tasks.

- Read the relations from the text file "input.txt". You can use your code from HW2 if you implemented to read the file. If you didn't implement it, please check HW2 to learn how to read the relations from the file.
- Determine each relation in "input.txt" whether it is reflexive, symmetric, anti-symmetric and transitive with your algorithm from HW2.
- In order to draw Hasse diagram, each relation must be POSET. Hence, the relation obeys the following rules:
  - Reflexivity
  - Anti-symmetric
  - Transitivity

  If the relation is not a POSET, your algorithm is responsible to CONVERT it to POSET.
  - If the relation is not reflexive, add new pairs to make the relation reflexive.
  - If the relation is symmetric, remove some pairs which make the relation symmetric. For instance, if the relation has (a, b) and (b, a), remove one of them randomly.
  - If the relation is not transitive, add new pairs which would make the relation transitive.
- After the relation becomes POSET, your algorithm should obtain Hasse diagram of the relation and write the diagram with the following format.
  - An example of the output format is given in "exampleoutput.txt". The file has the result of the first relation in "input.txt".
  - In "output.txt", each new Hasse diagram starts with "n".
  - The relation is (a, a), (a, b), (a, e), (b, b), (b, e), (c, c), (c, d), (d, d), (e, e)
  - The relation is already a POSET so we don't need to add or remove any pairs.
  - After "n", write the POSET in the next line as it is shown in "exampleoutput.txt".
  - Since the relation is POSET, it becomes (a, b), (b, e), (c, d) after removing reflexive and transitive pairs.
  - The following lines give each pair of Hasse diagram.
- You can implement your algorithm in Python, Java, C or C++.
- **Important:** Put comments almost for each line of your code to describe what the line is going to do.
- You should put your source code file (file name is problem1.{*.c*, *.java*, *.py*, *.cpp*}) and output.txt into your homework zip file (check Course Policy).