CS201

Mathematics For Computer Science

Indian Institute of Technology, Kanpur

End-Sem

Exam

Group Number: 5

Devanshu Singla (190274), Sarthak Rout (190772), Yatharth Goswami (191178)

Date of Submission: December 22, 2020

Question 1

Define two classes of n-variate polynomials as:

$$\begin{aligned} P_d(x_1, x_2, \ldots, x_n) &= \sum_{\substack{0 \le i_1, i_2, \ldots, i_n \le 1 \\ i_1 + i_2 + \cdots + i_n = d}} x_1^{i_1} x_2^{i_2} \cdots x_n^{i_n} \\ Q_d(x_1, x_2, \ldots, x_n) &= \sum_{\substack{0 \le i_1, i_2, \ldots, i_n \le d \\ i_1 + i_2 + \cdots + i_n = d}} x_1^{i_1} x_2^{i_2} \cdots x_n^{i_n} \end{aligned}$$

for all $d \geq 0$. Prove that:

$$\sum_{0 \le d \le r} (-1)^d P_d \cdot Q_{r-d} = 0$$

for all $r \geq 1$.

Solution

Consider the polynomial $F(y) = \prod_{i=1}^{n} (1 - x_i y)$.
It can be clearly seen that coefficient of y^k is the sum of all such terms obtained by multiplying the second term of each i-th factor $(1 - x_i y)$ which is $-x_i y$, which is linear in y, from any k mono-polynomials being multiplied and first term i.e. 1 from rest of the mono-polynomials.
Note: Here, a mono-polynomial is a factor in the expression for $F(y)$. Ex: $(1 - x_2 y)$

$$\therefore \text{ coefficient of } y^k \text{ in } F(y) = \sum_{\substack{0 \le i_1, i_2, \ldots, i_n \le 1 \\ i_1 + i_2 + \cdots + i_n = k}} (-x_1)^{i_1} (-x_2)^{i_2} \ldots (-x_n)^{i_n}$$

$$
\begin{aligned}
&= (-1)^{i_1+i_2+\cdots+i_n} \sum_{\substack{0\le i_1,i_2,\ldots,i_n\le 1 \\ i_1+i_2+\cdots+i_n=k}} x_1^{i_1}x_2^{i_2}\ldots x_n^{i_n} \\
&= (-1)^k P_k(x_1,x_2,\ldots,x_n)
\end{aligned}
$$

Functional equation for $\frac{1}{F(y)}$:

$$
\begin{aligned}
\frac{1}{F(y)} &= \prod_{i=1}^{n} \frac{1}{1-x_i y} \\
&= \prod_{i=1}^{n} \sum_{j\ge 0} (x_i^j y^j) \\
&= \sum_{k\ge 0} \Big(\sum_{\substack{0\le i_1,i_2,\ldots,i_n\le k \\ i_1+i_2+\cdots+i_n=k}} \prod_{r=1}^{n} x_r^{i_r} \Big) y^k \\
&= \sum_{k\ge 0} Q_k(x_1,x_2,\ldots,x_n) y^k
\end{aligned}
$$

Multiplying both of these functional equations,

$$
\begin{aligned}
F(y)\left(\frac{1}{F(y)}\right) &= \Big(\sum_{k\ge 0} (-1)^k P_k(x_1,x_2,\ldots,x_n) y^k\Big)\Big(\sum_{k\ge 0} Q_k(x_1,x_2,\ldots,x_n) y^k\Big) \\
1 &= \sum_{r\ge 0} \Big(\sum_{d=0}^{r} (-1)^d P_d(x_1,x_2,\ldots,x_n) Q_{r-d}(x_1,x_2,\ldots,x_n) \Big) y^r
\end{aligned}
$$

Equating powers of y on both sides,

$$
\begin{aligned}
\implies \quad & \sum_{d=0}^{r} (-1)^d P_d(x_1,x_2,\ldots,x_n) \cdot Q_{r-d}(x_1,x_2,\ldots,x_n) = 0, \text{ for } r \ge 1 \\
\implies \quad & \sum_{d=0}^{r} (-1)^d P_d \cdot Q_{r-d} = 0, \text{ for } r \ge 1
\end{aligned}
$$

Question 2

The algorithm in Lecture 18 for finding a perfect matching is wrong. Find a counter example, i.e., a bipartite graph on which the algorithm fails. Fix the algorithm by suitably modifying the definition of subgraph H.

Solution

Counter-Example:

All vertices are marked in bold and the sets $N(\{\cdot\}), \pi(\cdot)$ etc. are as defined in lectures. Consider the bipartite graph below G of 6 vertices which are labelled as $V_1 = \{$**1, 2, 3**$\}$ and $V_2 = \{$**a, b, c**$\}$ in each partition.

At a cursory glance, we can observe that a perfect matching will not exist as **1** and

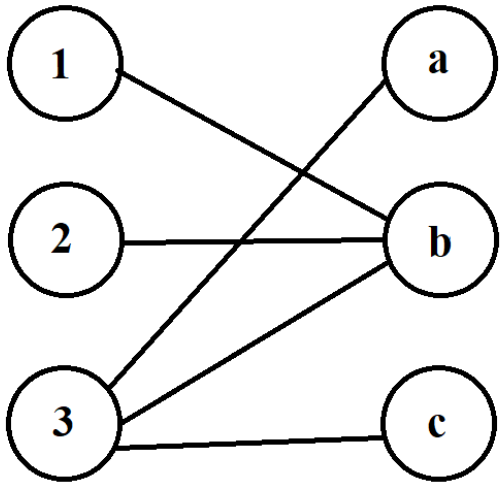


Figure 1: A bipartite graph G with edges

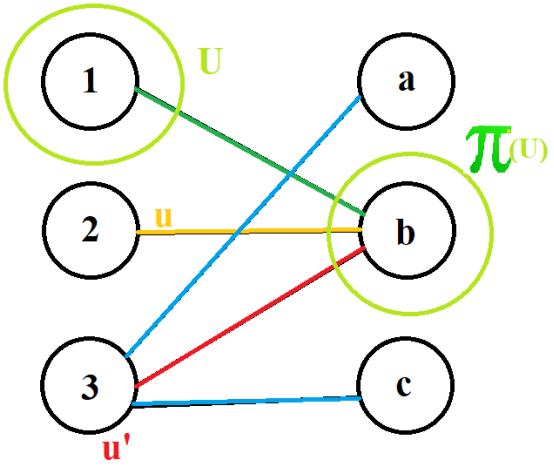
2 both have only edge with the same vertex **b**. Let us run the algorithm as described in the lecture to find a perfect matching on this graph.

First Iteration

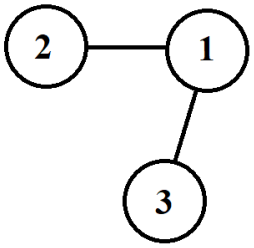
Initially, the set $U = \{\phi\}$ is empty and we choose vertex **1** as the next vertex u to be added, and we pair it with the only available vertex of $N(\{\mathbf{1}\}) = \{\mathbf{b}\}$ which is **b**.So, $\pi(\mathbf{1}) = \mathbf{b}$. This completes the first iteration of the algorithm and we get a perfect matching for subset $U \cup \{u\} = \{\mathbf{1}\}$. Also, the new U becomes $U \cup \{u\} = \{\mathbf{1}\}$.

Second Iteration

Next, we choose vertex **2** as our next choice for u. But, we see that $N(\{\mathbf{2}\}) = \{\mathbf{b}\} \subseteq \pi(U)$. Hence, we construct a sub-graph H as defined in the lecture to correct this situation if possible.



(a) Second Iteration of the Algorithm



(b) Sub-graph H

$H = (V_1, E_H)$ is defined as the collection of all such edges where atleast one vertex belongs to U and both vertices share an edge having a common vertex in $\pi(U)$.
As $U = \{\mathbf{1}\}$ and $\pi(\mathbf{1}) = \mathbf{b}$, we find (**2, 1**) and (**3, 1**) two such pairs of vertices $\in V_1$ both of which share a common vertex **2** in $\pi(U)$. We construct H as shown in the figure to the right above. Incidentally, the graph H here is a tree T in itself. It is also a spanning tree in its spanning forest.
Now, $T = \{\mathbf{1, 2, 3}\}$ and $N(T) = \{\mathbf{a, b, c}\}$. As $|N(T)| = |T| > |T| - 1$, the algorithm guarantees that a perfect matching of $U \cup \{u\} = \{\mathbf{1, 2}\}$ shall exist. Next, we find a vertex u' such that $N(\{u'\}) \not\subseteq \pi(U)$ which is vertex $u' = \mathbf{3}$. The path from **2** to **3** is through **1**, so, we must interchange the edges between the pairs of vertices { (**2, 1**), (**1, 3**) }.
Without a loss of generality for $\pi(\mathbf{3})$, we set $\pi(\mathbf{3}) = \mathbf{a}$, $\pi(\mathbf{1}) = \mathbf{b}$ and $\pi(\mathbf{2}) = \mathbf{b}$.

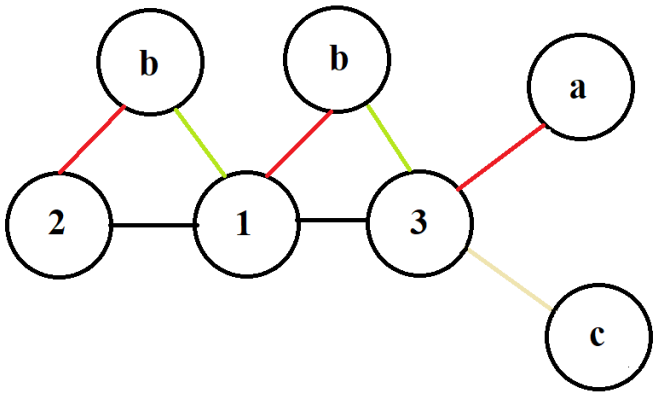


Figure 3: The change of mappings of π in this iteration: Red - new, Green - old

The iteration step ends here, but, we can clearly see that $\pi(\mathbf{1}) = \pi(\mathbf{2}) = \mathbf{b}$ which is not a perfect matching for $U \cup \{u\} = \{1, 2\}$ as required. So, the algorithm fails in the second iteration for this graph and the invariant that we have a perfect matching for the subset U is not maintained.

The Fixed Algorithm:

It can be observed above that the reason, the algorithm failed above is due to the inclusion of **3** in H, which was a previously unmatched vertex.
Actually, H should contain u and vertices of U **only**, hence, H should be redefined as $H = (U \cup \{u\}, E_H)$ instead of V_1.
Particularly, perfect matching for subset U is guaranteed by induction hypothesis and we are trying to add a new element u to the set. The final U will become $U \cup \{u\}$ after the completion of the iteration step. But, the mapping of the vertices of the new U are chosen from a graph H which was originally having all vertices $\in V_1$.

At any iteration, it is possible that the a path from u ends in $u' \in V_1$ such that $u' \notin U$ which was added in H as it had some common vertex in $\pi(U)$ with some previous vertex. But, during the iteration step, we are considering it as a matched vertex in U having an edge to $\pi(U)$ and trying to change it's mapping which is clearly making our invariant invalid.
Also, addition of extraneous edges of the unwanted u' bypasses the check $N(T) >$

$|T| - 1$ for existence of perfect matching of subset U.
The main steps for one iteration of the **new algorithm** will be:

- Let $G = (V_1, V_2, E)$ be a bipartite graph with $|V_1| = |V_2| = n$.
- Let a bijection π represent a perfect matching from V_1 to V_2.
- Let π be defined for a subset U of V_1.
- Choose a new vertex u to be added to U.
- If there is some vertex in V_2 which $\notin \pi(U)$, then, match u to that vertex and continue to next iteration.
- Otherwise, $N(\{u\}) \subseteq \pi(U)$. Then, construct a sub-graph $H = (\mathbf{U} \cup \{\mathbf{u}\}, E_H)$.
- Add only those edges $(u_1, \pi^{-1}(u_2))$ to E_H iff $(u_1, u_2) \in E$ AND $\pi^{-1}(u_2)$ exists AND $u_1 \neq \pi^{-1}(u_2)$ (to avoid self loops).
- As described in the lecture, we can compute a spanning forest of H and we obtain a tree T rooted at U.
- If $N(T) \leq |T| - 1$, we stop and declare that there is no perfect matching possible.
- Otherwise, we will always find a vertex $u' \in T \cap U$, such that, it can be paired with some other element, say y.
- On the path to the vertex u', we exchange all the edges with the vertex pairs, such that, u gets mapped to an element in $\pi(U)$ and the element u' is mapped to that unmatched element y outside $\pi(U)$ effectively extending it.
- The iteration step is complete.

Question 3

Let G be connected graph on $n \geq 4$ vertices with $2n - 2$ edges. Prove that G has two cycles of equal length.

Solution

We will assume that graph $G = (V, E)$ been talked about above is a simple graph in addition to being connected.

Theorem 3.1. Graph G is a tree (connected and acyclic graph) iff every two nodes of G are joined by a unique path.

Proof. If we have a graph G, which is tree, then it must be connected and has no cycles. Since, G is connected, there must be atleast one simple path between any two vertices. If there are more than one path between two vertices, then parts of those paths can be joined to form a cycle. Thus, there must be exactly one path.

Now suppose we have a graph G with a unique path between any two vertices. Clearly, this graph is connected. If G contains a simple cycle, then there are two simple paths between the vertices in that cycle which contradicts our assumption. Hence, there can be no cycles and graph G is a tree. □

Definition 3.2. If G is a connected graph on n vertices, a **spanning tree** for G is a subgraph of G that is a tree on these n vertices. ◊

Theorem 3.3. Every connected graph has a spanning tree.

Proof. We will prove this by induction on number of edges. For the case of connected graph with 0 edges, it is simply a vertex and hence already a tree.
Now, suppose G has $p \geq 1$ edges. If G is a tree, we are done as it is it's own spanning tree. Otherwise, G contains a cycle (using **Theorem 3.1**). Remove one edge from this cycle. The resulting graph G' is also connected and has $p - 1$ edges which contains a spanning tree using induction hypothesis, which is also the spanning tree for the graph G. □

Theorem 3.4. If we add a edge in a tree T, then the resulting graph is no more acyclic.

Proof. Choose any two arbitrary vertices u and v in the tree, which are not connected directly. Since, the graph is a tree, there exists a simple path between them say P. Adding an edge (u, v) in the graph leads to another simple path between them. The path P along with edge (u, v) produces a cycle in the graph and hence it is no more acyclic. □

Now, using the above theorem we can see that for our original graph G, we will have a spanning tree T which is connected and acyclic by definition. We know that T contains $n - 1$ edges and we are left with $2n - 2 - n - 1 = n - 1$ edges which are present in G and not in T, let's call this set of edges as E'. Notice, that upon addition of any edge, say (u, v), from the set E', there would have existed at least 1 path between u and v (G is connected graph) before addition of edge say P, which after addition of an edge (u, v) forms a cycle with edge (u, v) and hence results in addition of at least 1 cycle to the original graph. Hence, we can obtain at least $n - 1$ different cycles by addition of edges from E' to T. But notice the fact that since the graph is on n vertices and simple in nature, the size of cycles can only be in range of 3 to n. Hence, there are $n - 2$ possibilities of sizes of cycles.

Therefore, the problem becomes distributing $n - 2$ sizes of cycles (holes) among at least $n - 1$ cycles (pigeons). Hence, using pigeon hole principle we can say that there exists atleast two cycles of same size.

Question 4

A completed Sudoku puzzle is a 9×9 grid filled in with numbers 1 to 9 according to the rules of Sudoku. We say two such puzzles are the same if one can be obtained from other by any of the following operations and their compositions:

- Rotation by 90, 180, and 270 degrees
- Flips along vertical, horizontal, and diagonal axes
- Rotation by 180 degree of each of the 3×3 subgrid simultaneously

Describe the subgroup made up of above three operations. Assuming total number of completed puzzles to be N, calculate the number of distinct completed puzzles.

Solution

Let a 9×9 grid of Sudoku be represented by matrix $A_{9\times 9}$. Let

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$$

where a_{ij} is a 3×3 matrix.

The operations given in question when applied to A results in rotations and flipping of a_{ij} also. Let the set of $n \times n$ matrices be given by $\mathbf{M_n}$ Let us define following operations on $\mathbf{M_3}$:

1. e: identity operation returns same matrix. ($e(a) = a$ where $a \in \mathbf{M_3}$)
2. r: rotation operation rotates matrix clockwise by $90°$ (maps element at (i, j) to $(j, 3 - i)$).
3. t: returns transposition of matrix. ($t(a) = a^T$).

The compositions of these operations will form set of transformations on $\mathbf{M_3}$. Let this set be represented by O_3 and let define binary operation $(.)$ on O_3 as composition or $a.b = a \circ b, \forall a, b \in O_3$.

Since after four 90° rotations, matrix comes back to original form $\implies r^4 = e$. Similarly, after transposing matrix 2 times, it comes back to same form $\implies t^2 = e$.

It can be easily verified through matrix transformations that $t = rtr = r^2tr^2 = r^3tr^3$. For example, in the case of rtr, let B be a 3×3 matrix, then $r(B)$ maps element at (i, j) in B to place $(j, 3 - i)$, then $(tr)(B)$ maps it to $(3 - i, j)$ and finally $(rtr)(B)$ map it to $(j, 3 - (3 - i)) = (j, i)$, which is mapping induced by transposition of B .Since e being identity operation $\implies a.e = e.a = a \forall a \in O_3$ and since $r^{4k} = e, \forall k \geq 0 \implies r^{4k}tr^{4k} = r^{4k+1}tr^{4k+1} = r^{4k+2}tr^{4k+2} = r^{4k+3}tr^{4k+3}, \forall k \geq 0 \implies r^ktr^k = t, \forall k \geq 0$.

$\therefore tr^l = (r^{3l})t(r^{3l})r^l = r^{3l}tr^{4l} = r^{3l}te = r^{3l}t$, for some $l \geq 0$. Also since $tr^lt = r^{3l}t.t = r^{3l}e = r^{3l}$, we can convert any sequence of r, t into the form of r^xt^y, for some $0 \leq x \leq 3, 0 \leq y \leq 1$ ($r^{4k} = e, \forall k \geq 0$). Since all elements of O_3 are just compositions of e, r, t, or sequence of r, t, hence all the elements of O_3 are r^xt^y s.t. $0 \leq x \leq 3, 0 \leq y \leq 1$.

By above discussion it can be easily seen that $(.)$ satisfies closure and associative properties. Also there exists identity element e as defined above s.t. $x.e = e.x = x, \forall x \in O_3$. Hence, $(.)$ also satisfies identity property. Now, for any element $o = r^xt^y \in O_3$ for $0 \leq x \leq 3, 0 \leq y \leq 1$, there exists $o^{-1} = r^{(4-x) \pmod 4}t^{(2-y) \pmod 2}$ s.t. $o.o^{-1} = e$. Hence, $(.)$ also satisfies inverse property. Since $(.)$ satisfies closure, associative, identity and inverse properties, it implies it form group with O_3.

Now, we will see how the rotations and transformations on 3×3 matrix can be represented through operations of set O_3:

1. No tranformation: e
2. 90° clockwise rotation: r
3. 180° clockwise rotation: r^2
4. 270° clockwise rotation: r^3
5. vertical flip: rt
6. horizontal flip: r^3t

7. left diagonal flip: t
8. right diagonal flip: r^2t

Now we will try to see the effect of doing transformations on 9×9 matrix A, on 3×3 sub-matrix of A i.e. a_{ij} for some $1 \leq i \leq 3, 1 \leq j \leq 3$:

Transformation 1: $90°$ clockwise rotation.
Rotation will map element at position (m, n) to $(n, 9 - m)$ in A. An element of $a_{i,j}$ at position (x, y) is at position $(3(i-1)+x, 3(j-1)+y)$ in A, for some $1 \leq x, y \leq 3$. Hence, after rotation it will get mapped to position $(3(j-1)+y, 9-3(i-1)-x)$.
Now, notice that rotation of 9×9 matrix as a whole can be seen as rotation of 3×3 matrix with $a_{i,j}$ as elements and then rotation of each individual $a_{i,j}$ matrix. To see this, first consider rotation of A with $a_{i,j}$ matrices as elements, then $a_{i,j}$ will map to $a_{j,3-i}$ and the element at (x, y) in $a_{i,j}$ will remain at same place (x, y) in $a_{j,3-i}$. Now, if each individual $a_{m,n}$ is rotated, then element at (x, y) in $a_{j,3-i}$ changes to position $(y, 3 - x)$. Hence, the final position of the element after the two rotations is $(3(j-1)+y, 3(3-i)+3-x) = (3(j-1)+y, 9-3(i-1)-x)$, same as when grid is normally rotated, $\forall i, j, x, y$.

Transformation 2: Transpose of matrix.
Transpose of A will map element at i, j to j, i. Like earlier, an element of $a_{i,j}$ at position (x, y) is at position $(3(i-1)+x, 3(j-1)+y)$ in A, for some $1 \leq x, y \leq 3$. Hence, after transposition it will get mapped to position $(3(j-1)+y, 3(i-1)+x)$.
Notice that transposition of 9×9 matrix as a whole can be seen as transposition of 3×3 matrix with $a_{i,j}$ as elements and then transposition of each individual $a_{i,j}$ matrix. To see this, first consider transposition of A with $a_{i,j}$ matrices as elements, then $a_{i,j}$ will map to $a_{j,i}$ and the element at (x, y) in $a_{i,j}$ will remain at same place (x, y) in $a_{j,i}$. Now, if we take transposition of each individual $a_{m,n}$ matrices, then element at (x, y) in $a_{j,i}$ changes to position (y, x). Hence, the final position of the element after the two transpositions is $(3(j-1)+y, 3(3-i)+x)$, same as when normal transposition of original grid is taken, $\forall i, j, x, y$.

Hence, if we represent transformation operation on 9×9 grid with $z = (x, y) \in O_3 \times O_3$ s.t. x represents transformation on main 9×9 matrix, A considering $a_{i,j}$ as elements and treating A as 3×3 matrix, and y represents further transformation on each $a_{i,j}$ matrices. Hence, the composition of two operations $z_1, z_2 \in O_3 \times O_3$ is given as

$z_1 \circ z_2 = (x_1, y_1) \circ (x_2, y_2) = (x_1.x_2, y_1.y_2)$. Define binary operation $(.)$ on $O_3 \times O_3$ as composition of operands or $z_1.z_2 = z_1 \circ z_2, \forall z_1, z_2 \in O_3 \times O_3$.

Now we will define operations on 9×9 grid as asked in question:

1. Rotation by 90, 180, and 270 degrees
   Let operation $R = (r, r)$ will represent rotation by 90° of the grid as has shown previously in discussions. Hence, $R^2 = (r^2, r^2)$ and $R^3 = (r^3, r^3)$ will represent 180 and 270 degree rotations respectively. Also, the identity operation $E = (e, e)$ will represent 0° rotation. Also, $R^{4k} = (r^{4k}, r^{4k}) = (e, e) = E, \forall k \geq 0$.

2. Flips along vertical, horizontal, and diagonal axes.
   Let the operation $T = (t, t)$ represent the left diagonal flip or transpose of grid. Then by similar logic as in O_3 operations, $RT = (rt, rt)$ represents vertical flip, $R^3T = (r^3t, r^3t)$ represents horizontal flip, $T = (t, t)$ represents left diagonal flip and $R^2T = (r^2t, r^2t)$ represents right diagonal flip. Also, $T^2 = (t^2, t^2) = (e, e) = E$.

3. Rotation by 180 degree of each of the 3×3 subgrids simultaneously.
   The operation $I = (e, r^2)$ represents the above sought transformation.

The set of transformations over 9×9 grid with composition of transformations as binary operator will form a group. Consider the set O_9 made up only of sequence of transformations of R, T, I and identity transformation E. By similar argument as in case of sequence of r and t transformations, all sequences of R and T will be of the form R^xT^y, for $0 \leq x \leq 3, 0 \leq y \leq 1$. Now consider following properties of operation I:

1. $I^2 = (e^2, (r^2)^2) = (e, r^4) = (e, e) = E$. Hence, $I^{2k} = E, \forall k \geq 0$.

2. For $k \geq 0$, $IR^k = (e, r^2).(r^k, r^k) = (r^k, r^{k+2}) = (r^k, r^k).(e, r^2) = R^kI$. Hence, I and R^k are commutative.

3. $IT = (e, r^2).(t, t) = (t, r^2t) = (t, tr^6) = (t, tr^2) = (t, t).(e, r^2) = TI$. Hence, I and T are commutative.

4. Since I is commutative with both T and R^k for $k \geq 0 \implies$ It is commutative with sequences of T and R also.

By the above properties, for any sequence of I, T and R, I being commutative to R and T, all I's instead of being multiplied in between sequences of T and R can be

multiplied at last leaving sequence of T and R in same relative order, which in turn will have form of R^xT^y, where $0 \leq x \leq 3, 0 \leq y \leq 1$. Since $I^2 = E \implies I^k = I^{k \pmod 2}$. Hence any sequence of I, T and R can only be of the form $R^xT^yI^z$, where $0 \leq x \leq 3, 0 \leq y \leq 1, 0 \leq z \leq 1$. Hence, O_9 consists of elements $R^xT^yI^z$, where $0 \leq x \leq 3, 0 \leq y \leq 1, 0 \leq z \leq 1$ and each element corresponds to unique transformation on 9×9 grid given in question.

It can be easily seen that $(.)$ in O_9 satisfies closure and associative properties as $(.)$ in O_3 satisfies both of these. Also E is identity element in O_9 since $Z.E = (z_1, z_2).(e, e) = (z_1, z_2) = Z = (z_1, z_2) = (e, e).(z_1, z_2) = E.Z$ for any $Z \in O_{9\times9}$. Hence, identity property is also satisfied. For any element $O = R^xT^yI^z$, where $0 \leq x \leq 3, 0 \leq y \leq 1, 0 \leq z \leq 1$ in O_9, there exists inverse element $O^{-1} = R^{(4-x) \pmod 4}T^{(2-y) \pmod 2}I^{(2-y) \pmod 2}$ s.t. $O.O^{-1} = E$. Hence, inverse property is also satisfied.
Since, $(.)$ in O_9 satisfies closure, associative, identity and inverse properties $\implies$ $(.)$ forms group in O_9.

Since O_9 is subset of set of transformation on 9×9 grid, hence composition operator with O_9 forms subgroup of group of transformations on 9×9 grid.

Since elements of subgroup O_9 can only be of the form $R^xT^yI^z$, where $0 \leq x \leq 3, 0 \leq y \leq 1, 0 \leq z \leq 1$, and therefore x can take 4 values, y can take 2 values and z can take 2 values independent of each other, hence by multiplicative principle, total number of elements in O_9 = $4 \times 2 \times 2 = 16$.
Since for a completed Sudoku grid A, $O(A)$ is same as $A, \forall O \in O_9$ and $|O_9| = 16$, hence total 16 equivalent Sudoku puzzles of each completed Sudoku puzzle are present in set of completed Sudoku puzzle with cardinality $N \implies$ total number of distinct completed Sudoku puzzles = $N/16$.

Question 5

Let $(G, \cdot)$ be a group. A proper subgroup of G is a subgroup which is a proper subset of G. H is a maximal subgroup of G if H is a proper subgroup of G and there is no other proper subgroup H' such that $H \subset H'$.
Give an example of a group that does not have a maximal subgroup. Under what conditions will G have a maximal subgroup?

Solution

There exists various examples of groups that do not have a maximal subgroup like the group $(\{e\}, \cdot)$ with e as the identity element of the group, $(\mathbb{Q}, +)$ and the Prüfer group. We will provide a proof of the group $(\mathbb{Q}, +)$ not having a maximal subgroup.

Proof. Suppose, M is a non-zero maximal subgroup of $\mathbb{Q}$ and let $x \in \mathbb{Q} \setminus M$ and $y \in M,\ y \neq 0$. The existence of such a y is guaranteed, since if there is no such y then it means that M only contains 0 in it and hence it is trivially not maximal.
Since, $x, y \in \mathbb{Q}$ therefore $\frac{y}{x} = \frac{c}{d}$ where c, d are integers. Therefore $c \neq 0$. Let's form another group $M' = \{m + nx | m \in M, n \in \mathbb{Z}\}$ with operation (+). To prove that $(M', +)$ forms a group, we will show closure and existence of inverse only, as associativity and existence of identity is justified because of elements of M' being part of $(\mathbb{Q}, +)$. For showing closure, take $a = m_1 + n_1 x$ and $b = m_2 + n_2 x$, where $m_1, m_2 \in M$ and $n_1, n_2 \in \mathbb{Z}$. Therefore $a + b = m_1 + m_2 + (n_1 + n_2)x$. Now, since $m_1 + m_2 \in M$ because of closure of $(M, +)$ hence let $m_1 + m_2 = m_3 \in M$ and $n_1 + n_2 = n_3 \in \mathbb{Z}$. Therefore, $a + b = m_3 + n_3 x$ which belongs to M'. For showing the existence of an inverse, choose and arbitrary $a = m + nx \in M'$ and $b = -m + (-n)x$. Now, $b \in M$ since, $-m \in M$ as existence of inverse for $(M, +)$ is guaranteed. Therefore $a + b = 0$ and since a was chosen arbitrarily, existence of inverse is guaranteed for M'. Hence, M' forms a group under the operation (+).
We will now prove that $\frac{x}{c} \in \mathbb{Q}$ does not belong to M'. Suppose, on contrary that it belongs to M', then $\frac{x}{c} = m + nx$ for some $m \in M$ and $n \in \mathbb{Z}$. Therefore, $x = cm + cnx = cm + ndy \in M$, which is a contradiction to the fact that $x \in \mathbb{Q} \setminus M$. Therefore, M' is a proper subgroup of $(\mathbb{Q}, +)$. Now, since $x \in M'$ and not in M, therefore M' is strictly larger than M under the partial order given by operation $\subseteq$ and it is also a proper subset of $(\mathbb{Q}, +)$ since $\frac{x}{c} \notin M'$. Hence, M can't be maximal. □

Characterisation for Finite Groups Let $(G, .)$ be a finite group. Let us define set A_G as the set of all perfect subgroups of G with $\subseteq$ as partial order on A_G.

Lemma 5.1. For group G, the set consisting only of identity element, $e \in G$, forms a subgroup in G.

Proof. Since $e.e = e \in \{e\}$, closure property is satisfied. Since $(e.e).e = e.(e.e)$, associative property is satisfied. Since e itself is identity element and is also inverse of itself $(e.e = e)$, hence identity and inverse properties properties are also satisfied.
Since closure, associative, identity and inverse are satisfied hence $\{e\}$ forms group. □

Since for every group G there exists subgroup $\{e\}$ (by lemma 5.1) $\implies A_G \neq \phi \implies$ there exists a proper subgroup $g \in A_G$ of G. If there is no maximal element of A_G w.r.t. partial order $\subseteq \implies$ for some $g \in A_G, \exists h \in A_G$ s.t. $g \subseteq g_1$ but since $g_1 \neq g \implies g \subset g_1 \implies |g| < |g_1| \implies |g_1| \geq |g| + 1$. $\therefore$ if $g_i \in A_G \implies \exists g_{i+1} \in A_G$ s.t. $g_i \subset g_{i+1}$ and $|g_{i+1}| \geq |g_i| + 1 \implies |g_i| \geq |g_1| + i - 1 \geq |g| + i$. For $i = |G|, |g_{|G|}| \geq |g| + |G| \geq |G|$ but $g_{|G|}$ being a proper subgroup of $G \implies |g_{|G|}| < |G|$
This contradiction implies that our assumption is wrong and there exist a maximal element in A_G. G being an arbitrary group it implies there exists a maximal proper subgroup in all finite groups.

Characterisation for General Groups Since it has not been explicitly mentioned that $(\cdot)$ operator in G need not to be commutative, we assume that it is commutative as told by sir in lecture.

Let $\mathbb{S}$ be set of subgroups of G. If $A \in \mathbb{S}$ be a subgroup in G, then denote the corresponding quotient group G/A by Q_A.

Claim: G has no maximal proper subgroup iff there exist no subgroup in G for which its quotient group is finite. (Here quotient group being finite also include the condition it does not contain $[e]$ only as it is the case of subgroup being the whole group itself instead of a proper subgroup)

Proof. ($\Leftarrow$)
Let there exist no subgroup in G for which quotient group is finite. Let us assume there exist a maximal subgroup $H \in G$. Since H is proper subgroup $\implies \exists a \in G$ s.t. $a \notin H$. Let $\langle H, a\rangle$ denote the smallest subgroup in G containing elements of H and a. If $\langle H, a\rangle \subset G$, then since $H \subset \langle H, a\rangle$, it will contradict the fact that H is maximal subgroup. Hence, $\langle H, a\rangle = G$.
Consider the set $H' = \{h.a^x | h \in H, x \in \mathbb{Z}\}$, for any two elements $h'_1 = h_1 a^{i_1}$ and $h'_2 = h_2 a^{i_2}$ in H', $h'_1.h'_2 = (h_1 a^{i_1}).(h_2 a^{i_2}) = (h_1.h_2)(a^{i_1+i_2}) \in H'$ since $h_1.h_2 \in H$ (closure property) and $i_1 + i_2 \in \mathbb{Z}$. Hence, closure property is satisfied in H'. Also for any element $h' = ha^i \in H' \exists (h')^{-1} = h^{-1}.a^{-i}$ s.t. $h'.(h')^{-1} = e$, where e is the identity element of G. Hence, inverse property is also satisfied. Associative and identity properties are also satisfied trivially by the respective properties of G since $H' \in G$. Hence H' forms subgroup in G. Since H' contains H and $a \implies \langle H, a\rangle \subseteq H' \implies G \subseteq H'$, but since H' is subgroup of $G \implies H' \in G$. Hence, $H' = G = \langle H, a\rangle$.
Let the equivalence class in Q_H containing $g \in G$ be $[g]$. Consider the following two cases:
case 1: There exist $g \in G$ s.t. $a^x = a^y = g$ s.t. $x > y$ where $x, y \in \mathbb{Z}$.
Let $k = x - y$, then $a^k = a^{x-y} = a^x.a^{-y} = g.g^{-1} = e$. Let $I = \{i \in [0, k-1] | a^i \notin H\}$. Clearly by definition of I, $|I| < k$. Since $G = H'$, all elements in G are of the form ha^i where $i \in \mathbb{Z}$. Since $a^k = e \implies a^{xk+i} = a^i, \forall x \in \mathbb{Z}, i \in [0, k-1]$ and also $a^i \in H, \forall i \notin I$, therefore all elements in G are of the form ha^i where $i \in I$. The set $H'_i = \{ha^i | h \in H\}$ must be subset of $[a_i]$ for all $i \in I$ as $\forall ha^i \in H'_i, \exists x = h \in H$ s.t. $x.a^i = ha^i$ and hence by definition of equivalence class, $ha^i \in [a^i]$. Since $G = \sum_{i \in I} H'_i \subseteq \sum_{i \in I}$ and G is union of all equivalence classes in Q_H, it implies union of all equivalence classes is subset of equivalence classes $[a_i], \forall i \in I \implies$ union of all equivalence classes $= \sum_{i \in I} [a_i]$. But since equivalence classes are disjoint $\implies$ number of equivalence classes $\leq k \implies Q_H$ is finite which is contradiction to our assumption. Hence, this case is not possible.

case 2: $\forall g \in G, \nexists x, y \in \mathbb{Z}$ and $x > y$ s.t. $a^x = a^y = g$.
Let us assume $a^i \in H, \forall i > 1$. Consider a^m, a^{m+1} for some $m > 1 \implies a^m, a^{m+1} \in H$ by our assumption. Since $a^m \in H$ and H being a subgroup in G, unique inverse of a^m in G, a^{-m} must also be inverse in H and hence $a^{-m} \in H$. By closure property, $a = a^{m+1}.a^{-m} \in H$ which is clear contradiction to the fact that $a \notin G$. Hence, our

assumption is wrong and $\exists k > 1$ s.t. $a^k \notin H$. Let $b = a^k$. Since $b \notin H$, it can be proved similar to the case of a that $H_b = \{h.b^x | h \in H, x \in \mathbb{Z}\}$ forms a subgroup in G. Since, $\forall h' = hb^x \in H_b, h' = ha^{kx} \in H' \implies H_b \subseteq H'$. $a \notin A_b$ because if $a \in A_b \implies a = hb^x$ or $a = a^{kx}$ but since $k > 1 \implies kx \neq 1, \forall x$ and $a^1 \neq a^y, \forall y \in \mathbb{Z}$ (by condition of this case) hence $a \neq a^{kx}, \forall x \in \mathbb{Z} \implies a \notin A_b$. Since $a \in H'$ but $a \notin A_b \implies A_b \subset H'$ or A_b is proper subgroup of G. Also since $b \in A_b$ but $b \notin H \implies H \subset A_b$ which is clear contradiction to the fact that H is maximal subgroup. Hence, this case is also not possible.

Since, both of the cases are not possible, it implies our initial assumption is wrong and hence, there does not exist any maximal subgroup in G.

($\Rightarrow$)
We will prove it by proving its contrapositive statement i.e. if their exists a finite quotient group in G then G has a maximal proper subgroup.
Let Q_A be the finite quotient group in G corresponding to subgroup A. If A is maximal proper subgroup then the statement is proved. So, consider the case that A is not a maximal proper subgroup. Then there exist a proper subgroup A_1 in G s.t. $A \subset A_1$. If $a, b \in G$ are equivalent w.r.t. A i.e. $\exists h \in A$ s.t. $a.h = b$, then they are also equivalent in A_1 since $h \in A \implies h \in A_1$. Hence, every equivalent class in Q_A must be a subset of a equivalence class in Q_{A_1}. Since A_1 is equivalence class of itself (generated by e) hence must be composed of an equivalence class in Q_A other than A because $A \subset A_1$. Since equivalence classes are disjoint and every equivalence class in $Q_(A_1)$ being superset of equivalence class in Q_A with one equivalence class (A_1) s.t. it is atleast composed of 2 equivalence classes in $Q_A \implies$ number of equivalence classes in $Q_(A_1)$ is less than that in Q_A.
Hence for every non-maximal proper subgroup A_i, there exists proper subgroup A_{i+1} s.t. $A_i \subset A_{i+1}$ and $|Q_{A_i}| > |Q_{A_{i+1}}|$ or $|Q_{A_i}| \geq |Q_{A_{i+1}}| + 1$. Let $n = |Q_A|$. Let us assume Q_{A_i} is not a maximal group $\forall i > 0$. $|Q_A| \geq |Q_{A_1} + 1| \geq |Q_{A_{n+1}} + n + 1| \geq n + 1 > n$, but $|Q_A| = n$ which is clear contradiction. Hence, there will exist i s.t. Q_{A_i} is maximal proper subgroup.
Hence, if their exists a finite quotient group in G then G has a maximal proper subgroup. □

Hence, only under the condition of existence of finite quotient group, there will exist a maximal proper subgroup of the group(commutative group only).