**Vipul Krishna**

**Assignment 7**

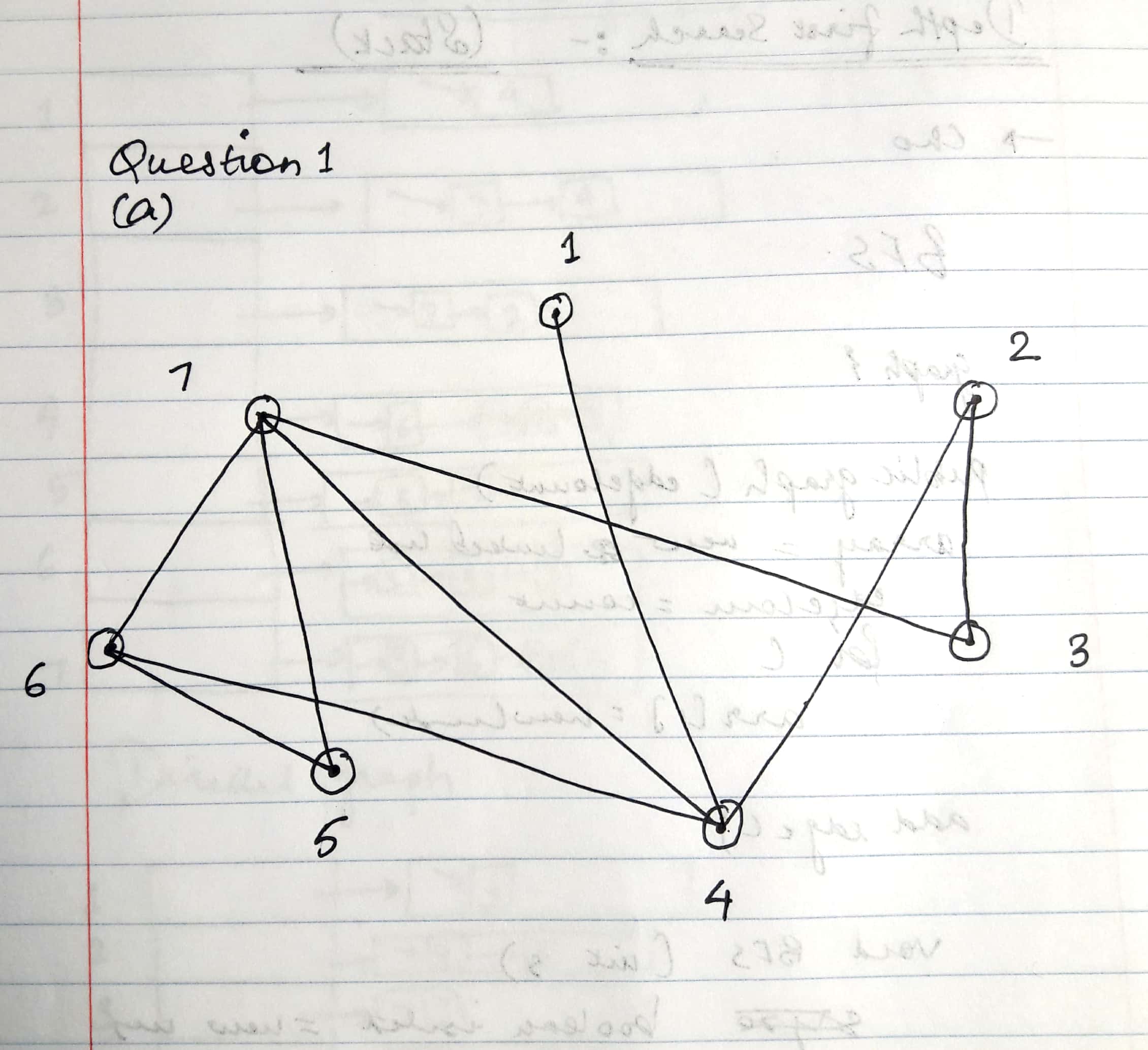
**INFO 6205 – Program Structures and Algorithms**

Question 1:

Vertices: 1, 2, 3, 4, 5, 6, 7

Edges:  (5, 6), (4, 6), (3, 7), (6, 7), (5, 7), (1, 4), (2, 4), (2, 3), (4, 7)

a)



b)

Yes, the graph is connected. This is because there exists a path between each pair of vertices. We can walk from any one vertex to any other using the edges.

The graph is not complete because we do not have a direct path between each pair of vertices. A complete graph is a graph in which each pair of vertices are connected by one edge directly. It is not the case here.

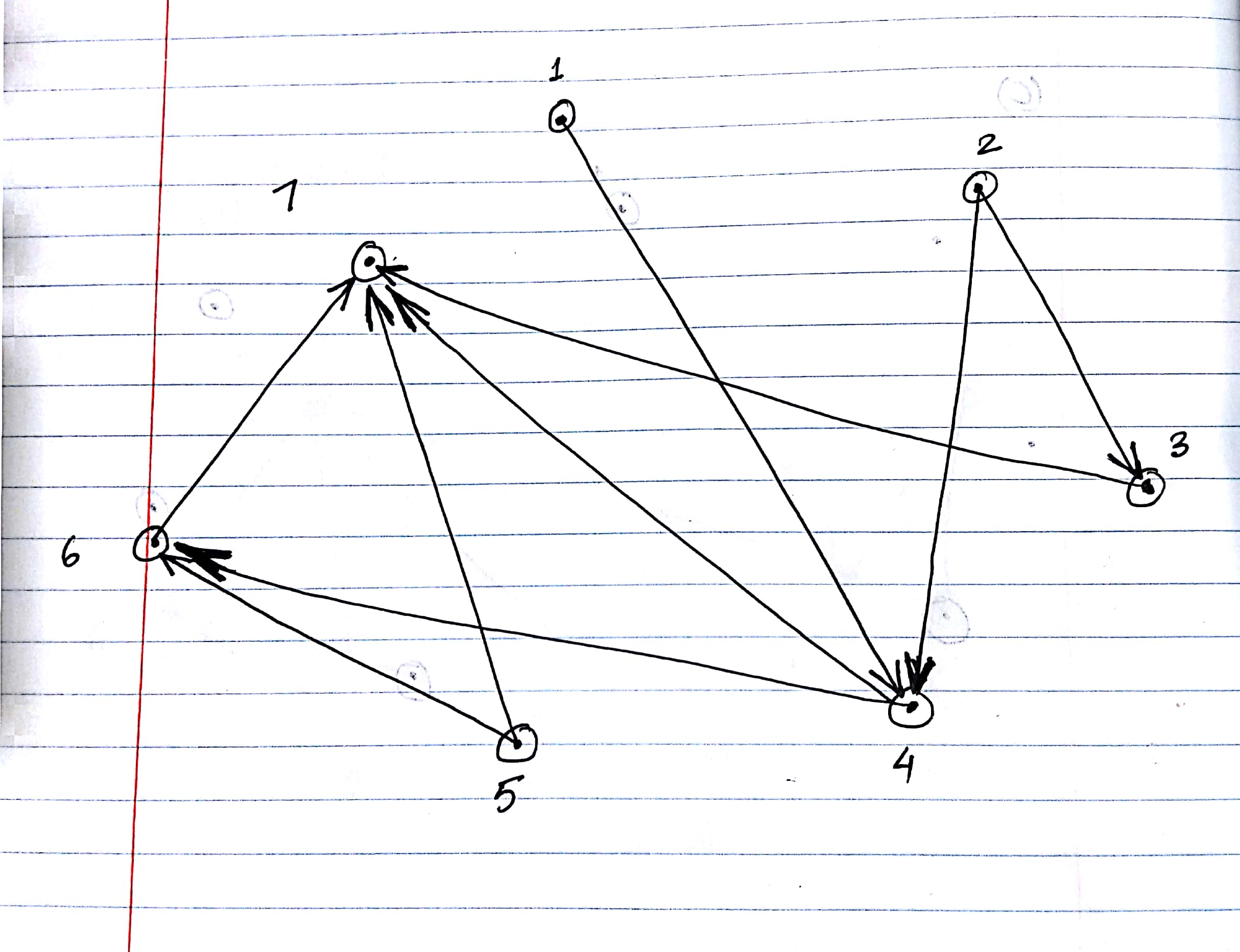
In case of an undirected graph, a graph will be complete only if the total number of edges of the graph is (N\*(N-1))/2. Here N is the number of vertices.

In this graph, number of edges = 9

Expected number of edges in case of complete graph with 7 vertices = 7\*6/2 = 21.

Thus, the graph is not complete.

c)



d)

No, the directed graph is not connected. This is because we do not have a path between each pair of vertices. However, this directed graph is **WEAKLY CONNECTED.**

The graph is not complete because we do not have a direct path between each pair of vertices. A complete graph is a graph in which each pair of vertices are connected by one edge directly. It is not the case here.

In case of a directed graph, a graph will be complete only if the total number of edges of the graph is (N\*(N-1)). Here N is the number of vertices.

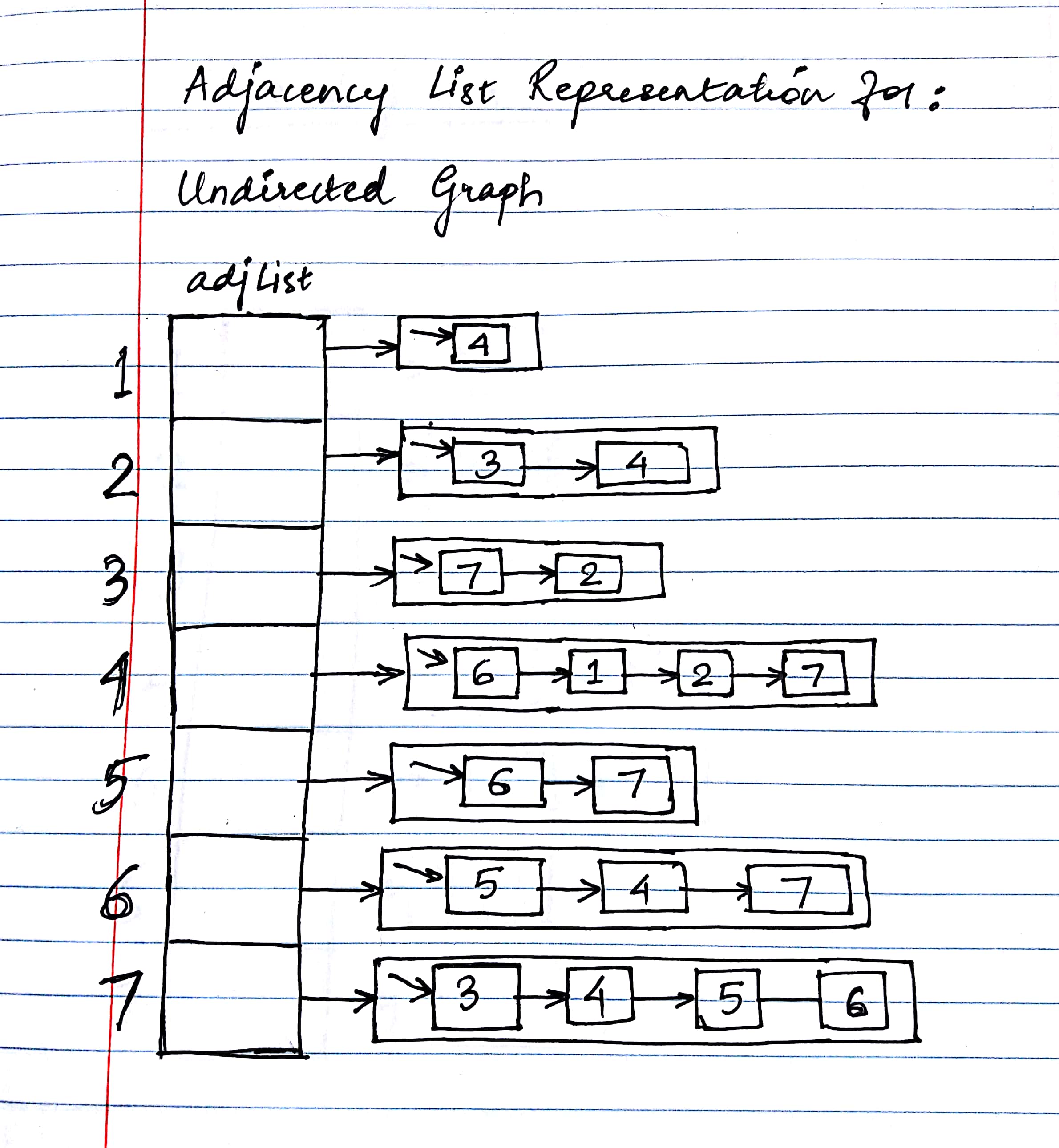
In this graph, number of edges = 9

Expected number of edges in case of complete graph with 7 vertices = 7\*6 = 42.

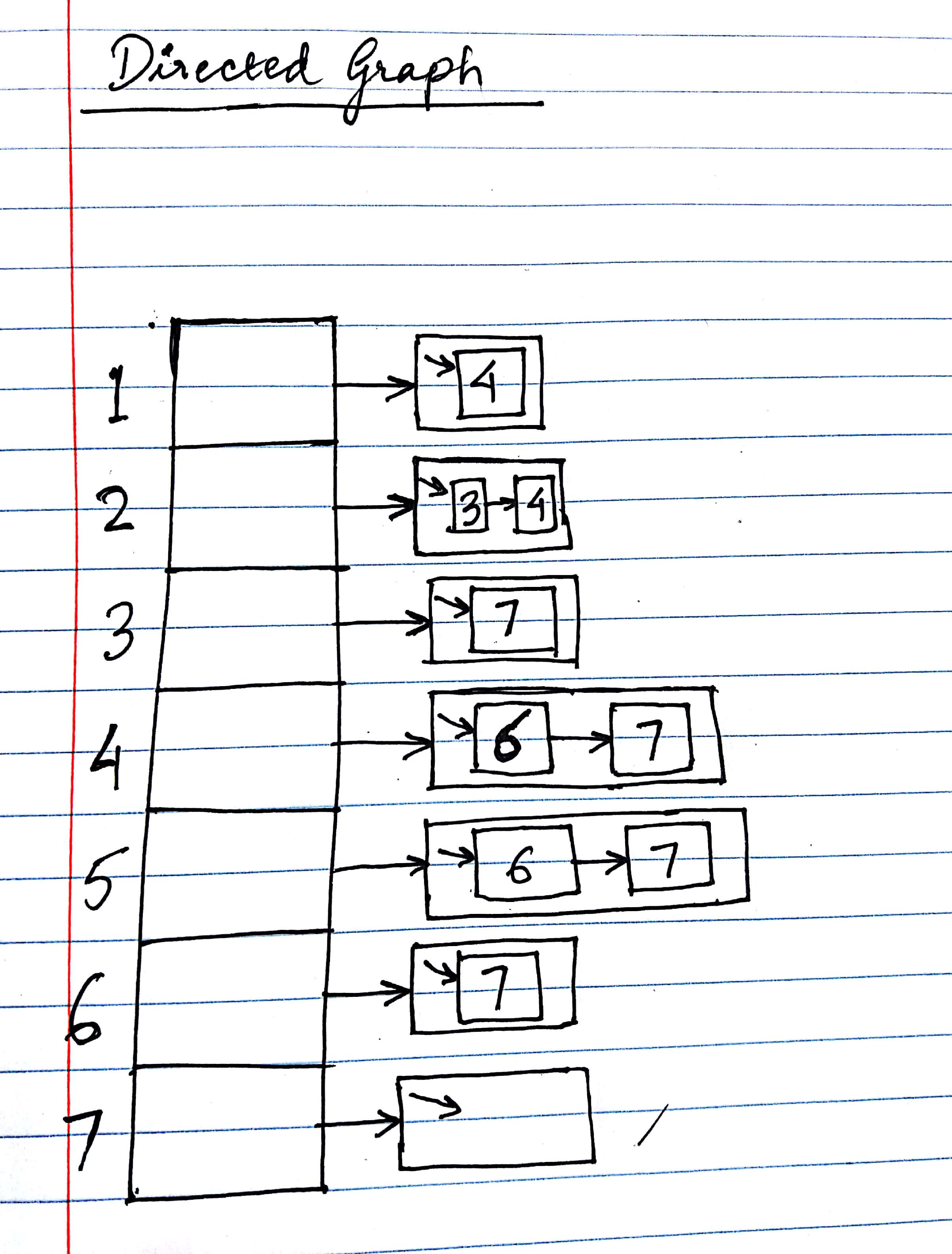
Thus, the graph is not complete.

Question 2:

Undirected Graph



Directed Graph



Question 3:

Java Project attached.

Question 4:

The space consumed by Adjacency List Representation of graph = O (|V|+ |E|) where V and E are the number of vertices and edges in the graph respectively.

Time complexity = O (|V|+ |E|) where V and E are the number of vertices and edges in the graph respectively.

Question 5:

1. Depth First Traversal starting at vertex 1 for the above graph:

Undirected Graph:

1, 4, 2, 3, 7, 5, 6

1. Java Project Attached
2. The running time complexity of the algorithm = O(V+E) where V is the number of vertices and E is the number of edges in the graph

Question 6

1. Breadth First Traversal starting at vertex 1 for the above graph:

Undirected Graph:

1, 4, 2, 7, 6, 3, 5

1. Java Project Attached
2. The running time complexity of the algorithm = O(V+E) where V is the number of vertices and E is the number of edges in the graph

Question 7

There are seven types of DNA mutations. They are as follows:

1. Missense Mutation - This is a substitution mutation in which one DNA base pair is replaced by another.

Example: ACTGGAC 🡪 ACTGGAT

1. Nonsense Mutation – This is like missense mutation but in this case, the new formed DNA sequence directs the cells to stop producing amino acids. So, this may result in formation of protein which might not work as expected.

Example: ACTGGAC 🡪 ACTGGAT

1. Insertion – Insertions are mutations in which a new base is added to the DNA sequence.

Example: ACTG 🡪 ACACTG

1. Deletion – This is the type of mutation in which a subsequence of DNA is completely removed from the DNA sequence.

Example: ACACTG 🡪ACTG

1. Frameshift Mutation – The proteins in the DNA have a length of 3-bases. Any change in the DNA sequence by insertion, deletion etc. keeps the length of the codon to 3-bases but the parsing of the DNA becomes incorrect because of the change in the original sequence. This is called frameshift mutation.

Example: CAT GAC TGA 🡪 ATG ACT GA (The first base C is removed)

1. Duplication – This is the case where a piece of DNA is repeated multiple times.

Example: ACTGACTG 🡪 ACTGACACACTG

1. Repeat Expansion - This is the case where the repeat occurs in a row in a much higher amount. The repeat is made up of 3-bases in case of a trinucleotide and 4-base when we have a repeating tetranucleotide.

Example of a trinucleotide: ACT GAC TAC ACT 🡪 ACT GAC TAC TAC TAC ACT

Question 8:

1. Tumor DNA Sequencing - Tumor DNA Sequencing is the test to identify and study the changes in DNA sequence in case of Cancer. Cancer is basically a genetic disease that is caused by the variation in the DNA sequence. These variations are caused mostly because of improper cell division or when cells are exposed to carcinogenic substances. Tumor DNA sequence helps us identify these variations.
2. The author considers an example of the EGFR gene. The mutation in these genes, called EGFR mutation, cause lung cancer. EGFR inhibitors can be used for treatment of lung cancer. This EGFR mutation can be found out using Tumor DNA Sequencing

Question 9:

De Brujin Graph algorithm is a way of representation of a DNA sequence by using its k-mer components. This algorithm works according to the following steps:

1. Choose a k-mer size
2. Split the given sequence into the chosen k-mer component
3. Construct a directed graph by connecting the k-mer pairs in such a way that there is an overlap between the last k-1 nucleotides of the first component and the first k-1 nucleotides of the next one
4. The arrow of the edge in the graph is directed towards the next component which has the first k-1 nucleotides overlapping

The k-mer selection is driven by the following factors:

1. The size of the genome under study
2. The memory available to store the DNA sequence
3. The reconstruction of genome under study. For smaller size k-mer, reconstruction of genome is difficult as compared to the higher size k-mer
4. The information which is required by following the algorithm. For lower k-mer values, there are greater chances of loss of information

Representation of De Brujin Graph:

Sequence Under Study = ATCGGACT

Consider a k-mer size of 4, i.e., k = 4;

So, the representation will be:

A T C G

T C G G

C G G A

G G A C

G A C T

**Directed Graph = ATCG 🡪 TCGG 🡪 CGGA 🡪 GGAC 🡪GACT**