

Module - 3.5: Practice Problems

1. Calculate time and space complexity of the following code snippet:

```
int a = 0, b = 0;
int matrix[N][M];
for (i = 0; i < N; i++) {
    a = a + i;
    matrix[i][i] = a;
}
for (j = 0; j < M; j++) {
    b = b + 2 * j;
}
```

2. Calculate time and space complexity of the following code:

```
int a = 0;
for (i = 0; i < N; i++) {
    for (j = N; j > i; j--) {
        a = a + i + j;
    }
}
```

3. Calculate time and space complexity of the following code:

```
int a = 0, i = N;
vector<int> vec;
while (i > 0) {
    a += i;
    i /= 2;
    vec.push_back(a);
}
```

4. Write a program to convert an **adjacency matrix** to an **adjacency list**. Similarly, write a program to convert an **adjacency list** to an **adjacency matrix**. Compare the time and space complexity of both programs. You can assume the graph is **undirected** and **unweighted**.

5. Write code to take the **edge list** of an **undirected unweighted** graph as input.
In one program turn the edge list to its **adjacency list** representation.
In another program turn it to its **adjacency matrix** representation.

Use this code snippet to take the graph as an input to your program:

<https://ideone.com/YeAUXe>