Assignment 2
Due date: 23 October 2024
TA: 薛凱駿, 楊承霖, 吳奇軒 (ECG 706)

1. (30%) Write C functions `read_matrix`, `print_matrix` and `search` that read triples (i.e., (row, col, value)) into a new sparse matrix, print_out the matrix and search for a value in a sparse matrix. Refer to the following sample driving `main()` function:

```
#define MAX_TERMS 101

int main {
     term mat[MAX_TERMS];
     read_matrix(mat); // read the input into the matrix
     print_matrix(mat); // print out the matrix
     search(mat, 2); // return 1 if found and 0 otherwise
     search(mat, -2); // return 1 if found and 0 otherwise
     return 0;
}
```

Sample Input:

```
2 3 4
0 0 1
0 1 3
1 1 -1
1 2 2
```

Sample Output:

```
1 3 0
0 -1 2
1
-1
```

2. (30%) Rewrite fast_transpose so that it uses ONLY ONE array rather than the two arrays required to hold row_terms and starting_pos.

   Note: You can only submit the C *function* (or *pseudo-code*).

3. (40%) Consider the railroad switching network (Figure below). Railroad cars numbered 0, 1, …, $n-1$ are the right. Each car is brought into the **stack** and removed **at any time**. For instance, if $n = 3$, we could move in 0, move in 1, move in 2, and then take the cars out, producing the new order 2, 1, 0. For $n = 4$, find out the IMPOSSIBLE permutations of the cars. Submit your answers as well as either your explanations or the C code.

   (*Hint*: Implement the stack and consider all sequences of *n* pushes and *n* pops.)