Implement Johnson Trotter algorithm to generate permutations

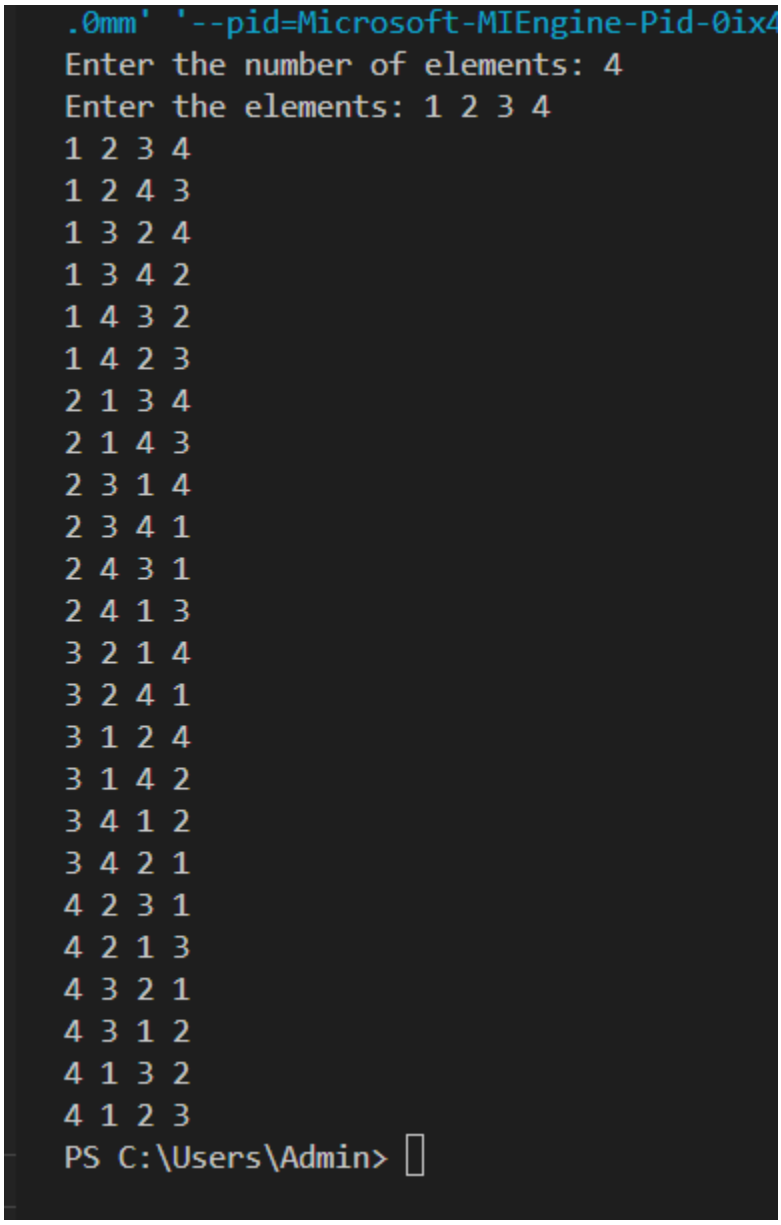
```c
#include <stdio.h>
```

```c
#include <stdlib.h>



void swap(int* a, int* b) {

    int temp = *a;

    *a = *b;

    *b = temp;

}



void generatePermutations(int arr[], int start, int end) {

    if (start == end) {

        for (int i = 0; i <= end; i++) {

            printf("%d ", arr[i]);

        }

        printf("\n");

    } else {

        for (int i = start; i <= end; i++) {
```

```
            swap(&arr[start], &arr[i]);

            generatePermutations(arr, start + 1, end);

            swap(&arr[start], &arr[i]); // backtrack

        }

    }

}



int main() {

    int n;

    printf("Enter the number of elements: ");

    scanf("%d", &n);



    int* arr = (int*)malloc(n * sizeof(int));

    if (arr == NULL) {

        printf("Memory allocation failed.\n");

        return 1;

    }



    printf("Enter the elements: ");
```

```
    for (int i = 0; i < n; i++) {

        scanf("%d", &arr[i]);

    }

    generatePermutations(arr, 0, n - 1);

    free(arr);

    return 0;

}
```

OUTPUT


.0mm' '--pid=Microsoft-MIEngine-Pid-0ix4
Enter the number of elements: 4
Enter the elements: 1 2 3 4
1 2 3 4
1 2 4 3
1 3 2 4
1 3 4 2
1 4 3 2
1 4 2 3
2 1 3 4
2 1 4 3
2 3 1 4
2 3 4 1
2 4 3 1
2 4 1 3
3 2 1 4
3 2 4 1
3 1 2 4
3 1 4 2
3 4 1 2
3 4 2 1
4 2 3 1
4 2 1 3
4 3 2 1
4 3 1 2
4 1 3 2
4 1 2 3
PS C:\Users\Admin>