Office Hour #6

```
// #1

//      0 1 2 3 4
// arr [4|0|0|1|3]
// Each array element is an index to an element of the array
// Nachenberg cycle of length n points to the next n elements then loops back to beginning

// Nachenberg cycle length of arr starting at cur
// Hint: modify array

int magicLength(int arr[], int cur);

int length(int arr[], int cur) {
    // base case
    if (arr[cur] == -1)
        return 0;

    int temp = arr[cur];
    arr[cur] = -1;

    int result = length(arr, temp) + 1;
    arr[cur] = temp;
    return result;
}
```

```
// #2

// print every even index forward, odd backward from linked list

void printEvenIndexForwardOddBW(Node* cur) {

    printh(cur, 0);

}

void printh(Node* cur, int index) {
    if (cur == nullptr)
        return;

    if (index % 2 == 0)
        cout << cur->value;

    printh(cur->next, index + 1);

    if (index % 2 == 1)
        cout << cur->value;
}
```