**Experiment No 3.2**

**Student Name: Ayushi Aggarwal UID: 21MCA2806**

**Branch: MCA Section/Group: 8B**

**Semester: 2nd Date of Performance: 17-04-2022**

**Subject Name: Computing Aptitude Subject Code: 21CAP-654**

1. **Aim/Overview of the practical:**
2. **Experiment 3.2.1**
3. You are given a linked list that contains N integers. You have performed the following reverse operation on the list:
4. • Select all the subparts of the list that contain only even integers. For example, if the list is {1,2,8,9,12,16}, then the selected subparts will be {2,8}, {12,16}.
5. • Reverse the selected subpart such as {8,2} and {16,12}.
6. Now, you are required to retrieve the original list.
7. Note: You should use the following definition of the linked list for this problem:
8. class Node {
9. Object data;
10. Node next;
11. }
12. Input format
13. • First line: N
14. • Next line: N space-separated integers that denote elements of the reverse list
15. Output format
16. Print the N elements of the original list.
17. Constraints
18. 1≤N≤103
19. 1≤Ai≤109
20. **Experiment 3.2.2**
21. Your task is to construct a tower in N days by following these conditions:
22. • Every day you are provided with one disk of distinct size.
23. • The disk with larger sizes should be placed at the bottom of the tower.
24. • The disk with smaller sizes should be placed at the top of the tower.
25. The order in which tower must be constructed is as follows:
26. • You cannot put a new disk on the top of the tower until all the larger disks that are given to you get placed.
27. Print N lines denoting the disk sizes that can be put on the tower on the ith day.
28. Input format
29. • First line: N denoting the total number of disks that are given to you in the N subsequent days
30. • Second line: N integers in which the ith integers denote the size of the disks that are given to you on the ith day
31. Note: All the disk sizes are distinct integers in the range of 1 to N.
32. Output format
33. Print N lines. In the ith line, print the size of disks that can be placed on the top of the tower in descending order of the disk sizes.
34. If on the ith day no disks can be placed, then leave that line empty.
35. Constraints
36. 1≤N≤106
37. 1≤size of a disk≤N

**Code:**

#include<stdio.h>

#include<stdlib.h>

 typedef struct node {

  int data;

  struct node \*next;

} n;

n\*start = NULL;

n\*head = NULL;

int count=0;

int nt,k;

void create\_node(int x) {

  n\*ptr = (n\*)malloc(sizeof(n));

  if(count==0) {

    ptr->data = x;

    ptr->next = start;

    start = ptr;

    head = ptr;

    count++;

  }

  else {

    ptr->data = x;

    ptr->next = NULL;

    head->next = ptr;

    head = ptr;

  }

}

void display\_list(n\*ptr) {

  if(ptr!=NULL){

  if(ptr->data % 2 == 0){

  n\*nptr = ptr;

  start = start->next;

  ptr=ptr->next;

  display\_list(ptr);

  printf("%d ",nptr->data);

  }

  }

}

 int main()

{

  scanf("%d",&nt);

  int ele;

  for(int i=0; i<nt; i++){

  scanf("%d",&ele);

    create\_node(ele);

    }

    while(start!=NULL){

    if(start->data % 2 == 0)

      display\_list(start);

    else{

    printf("%d ",start->data);

    start = start->next;

    }

    }

    return 0;

}

**Experiment 3.2.2**

/\*#include<stdio.h>

#include<malloc.h>

void Solve(int \*arr, int size) {

  /\*

   \*/

/\*}

int main(void) {

  int N;

  scanf("%d", &N);

  int \*arr = (int \*) malloc(sizeof(int) \* N);

  for (int i\_arr = 0; i\_arr < N; i\_arr++) {

    scanf("%d", &arr[i\_arr]);

  }

  Solve(arr, N);

}

Disk Tower problem in C language

1 year ago

1\*/

#include<stdio.h>

int main()

{

int disk, temp[100001] = {0};

scanf("%d", &disk);

int min = disk, size = disk;

int q;

for (int i = 0; i < disk; i++)

{

scanf("%d", &q);

temp[q] = q;

if(q == min)

{

while(temp[size])

{

printf("%d ", size);

size--;

}

min = size;

printf("\n");

}

}

}

**Output: -**

**3.2.1**



**3.2.2**

