S.No: 36	Exp. Name: ***Linked list Female gender first***	Date:2023-06-26

ID: 224G1A0591

2022-2026-CSE-B

Srinivasa Ramanujan Institute of Technology

Aim:

Consider a linked list consisting of name of a person and gender as a node. Arrange the linked list using ‘Ladies first’ principle. You may create new linked lists if necessary.
Note: Add node at the beginning.

Source Code:

rearrangeList.c

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

 struct Node
 {
    int data;
     char name[20];
      char gender;
       struct Node *next;
 };

  void segregateEvenOdd(struct Node **head_ref)
  {
    struct Node *end = *head_ref;
     struct Node *prev = NULL;
      struct Node *curr = *head_ref;

        while (end->next != NULL)
         end = end->next;

           struct Node *new_end = end;

             while (curr->data %2 != 0 && curr != end)
              {
                new_end->next = curr;
                 curr = curr->next;
                  new_end->next->next = NULL;
                   new_end = new_end->next;
              }

                if (curr->data%2 == 0)
                 {
                   *head_ref = curr;

                     while (curr != end)
                      {
                         if ( (curr->data)%2 == 0 )
                          {
                            prev = curr;
                             curr = curr->next;
                          }
                           else
                            {
```

```
                                prev->next = curr->next;

                                  curr->next = NULL;

                                    new_end->next = curr;

                                      new_end = curr;

                                        curr = prev->next;
                                }
                            }
                        }
                         else
                         prev = curr;
                          if (new_end!=end && (end->data)%2 != 0)
                           {
                             prev->next = end->next;
                              end->next = NULL;
                               new_end->next = end;
                           }
                            return;
}

 void push(struct Node** head_ref, char new_name[20], char new_gender)
 {
     struct Node* new_node = (struct Node*) malloc(sizeof(struct Node));
      strcpy(new_node->name, new_name);
       new_node->gender = new_gender;
        if (new_gender == 'F')
        new_node->data = 0;
         else if (new_gender == 'M')
         new_node->data = 1;
          new_node->next = (*head_ref);
           (*head_ref) = new_node;
 }
 void printList(struct Node *node)
 {
     while (node!=NULL)
      {
        printf("%s (%c)", node->name, node->gender);
         node = node->next;
          if (node!=NULL)
          printf(" --> ");
      }
 }

  int main()
  {
     struct Node* head = NULL;
      char name[20];
       char gender;
        int noOfInputs, i;
         int option;
          printf("Insert Data\n");
           do
            {
```


ID: 224G1A0591

```
                printf("Enter Name: ");
                 scanf(" %s", name);
                  printf("Enter Gender: ");
                   scanf(" %c", &gender);
                    push(&head, name, gender);
                     printf("1 : Insert into Linked List\n");
                      printf("0 : Exit\n");
                       printf("Enter your option: ");
                        scanf(" %d", &option);

            } while(option == 1);

              printf("Original Linked list \n");
               printList(head);
                segregateEvenOdd(&head);
                 printf("\nModified Linked list \n");
                  printList(head);
                   printf("\n");
                    return 0;
    }
```

Execution Results - All test cases have succeeded!

Test Case - 1
User Output
Insert Data Ganga
Enter Name: Ganga
Enter Gender: F
1 : Insert into Linked List 1
0 : Exit 1
Enter your option: 1
Enter Name: Yamuna
Enter Gender: F
1 : Insert into Linked List 1
0 : Exit 1
Enter your option: 1
Enter Name: Raj
Enter Gender: M
1 : Insert into Linked List 1
0 : Exit 1
Enter your option: 1
Enter Name: Veer
Enter Gender: M
1 : Insert into Linked List 1
0 : Exit 1
Enter your option: 1
Enter Name: Narmada
Enter Gender: F
1 : Insert into Linked List 1
0 : Exit 1
Enter your option: 1
Enter Name: Amar

Enter Gender: M
1 : Insert into Linked List 0
0 : Exit 0
Enter your option: 0
Original Linked list
Amar (M) --> Narmada (F) --> Veer (M) --> Raj (M) --> Yamuna (F) --> Ganga (F)
Modified Linked list
Narmada (F) --> Yamuna (F) --> Ganga (F) --> Amar (M) --> Veer (M) --> Raj (M)

Test Case - 2
User Output
Insert Data Ganga
Enter Name: Ganga
Enter Gender: F
1 : Insert into Linked List 1
0 : Exit 1
Enter your option: 1
Enter Name: Yamuna
Enter Gender: F
1 : Insert into Linked List 1
0 : Exit 1
Enter your option: 1
Enter Name: Narmada
Enter Gender: F
1 : Insert into Linked List 0
0 : Exit 0
Enter your option: 0
Original Linked list
Narmada (F) --> Yamuna (F) --> Ganga (F)
Modified Linked list
Narmada (F) --> Yamuna (F) --> Ganga (F)

Test Case - 3
User Output
Insert Data Raj
Enter Name: Raj
Enter Gender: M
1 : Insert into Linked List 1
0 : Exit 1
Enter your option: 1
Enter Name: Veer
Enter Gender: M
1 : Insert into Linked List 1
0 : Exit 1
Enter your option: 1
Enter Name: Amar
Enter Gender: M
1 : Insert into Linked List 0
0 : Exit 0
Enter your option: 0
Original Linked list

Amar (M) --> Veer (M) --> Raj (M)
Modified Linked list
Amar (M) --> Veer (M) --> Raj (M)