Data Structures and Algorithms

Lab Report

Lab04

Group Members Name & Reg #:	**Muhammad Haris Irfan** **(FA18-BCE-090)**
Class	Data Structures and Algorithms CSC211 (**BCE-3B**)
Instructor's Name	Dilshad Sabir

In Lab Tasks

Task:1

Debugging code for errors.

Solution:

I debugged the code and resolved the issues, the code had four syntax errors with the prototypes.

Task:2

Implementing Node Removal and Node Insertion Tasks

Solution

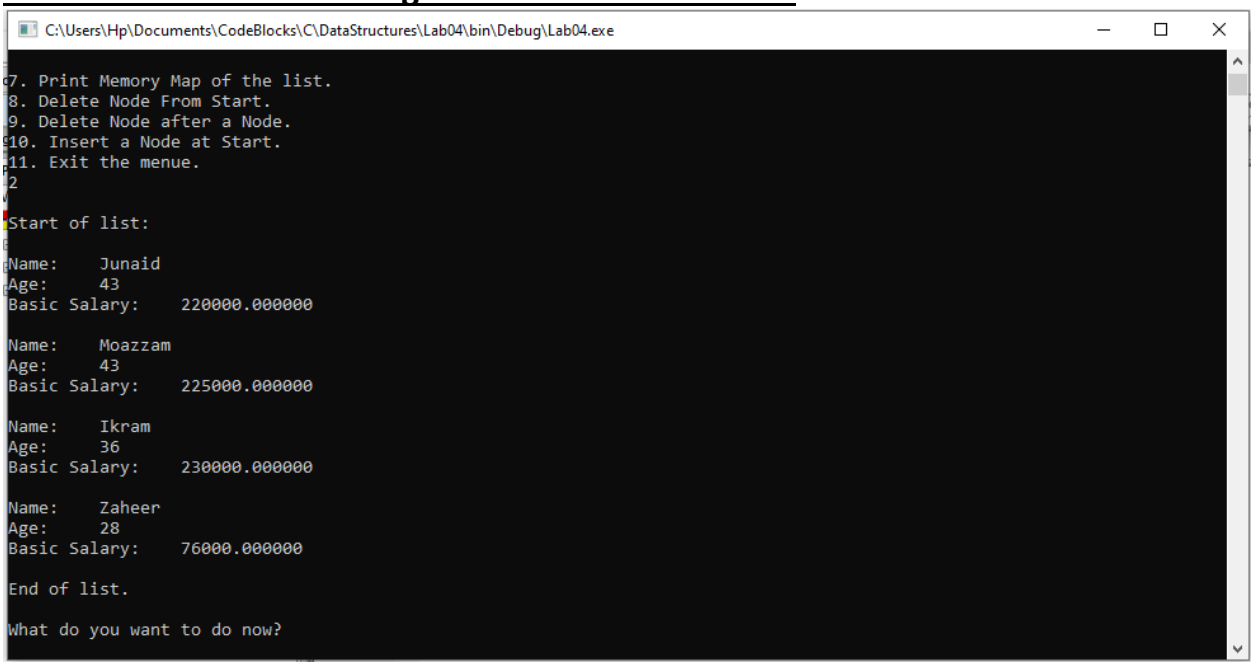
The code is shown below,

DELETING FROM THE BEGINING

The code is shown below,

```
void deleteNodeFromStart(struct node_d ** head)
{
    /*** To be completed by the students ***/
    struct node_d * move=*head;
    struct node_d * stat=*head;
    struct node_d * prevvv;
    while(move->next!=stat)
    {
        move=move->next;
    }

    move->next=stat->next;
    prevvv=stat->next;

    prevvv->prev=move;
    free(*head);
    *head=move->next;

        (*head)->prev=move;

    printf("\nNode from start deleted Successfully\n\n");
}
```

The Result of the following code is attached below:

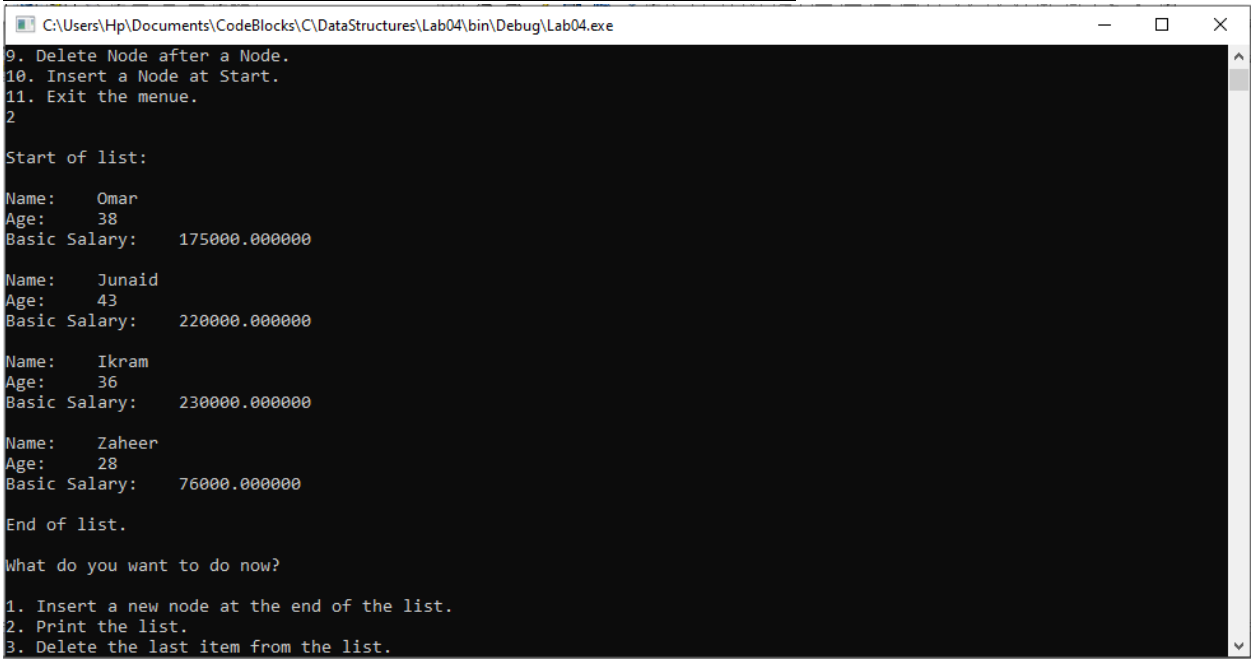


DELETING AFTER A NODE

The code is shown below,

```
int deleteNodeAfter(struct node_d * head, int idx)
{
   /*** To be completed by the students ***/
    int index=1;
    struct node_d *temp=head;
    struct node_d *del;
    struct node_d *a;
    struct node_d *b;
    while((temp->next!=NULL)&&(index!=idx))
    {
        temp=temp->next;
        index++;
    }


          del=temp->next;
          a=del->next;
          b=del->prev;


          a->prev=b;
          b->next=a;
     //head=temp->next->next;
     free(del);
    return(0);
}
```

The Result of the following code is attached below:

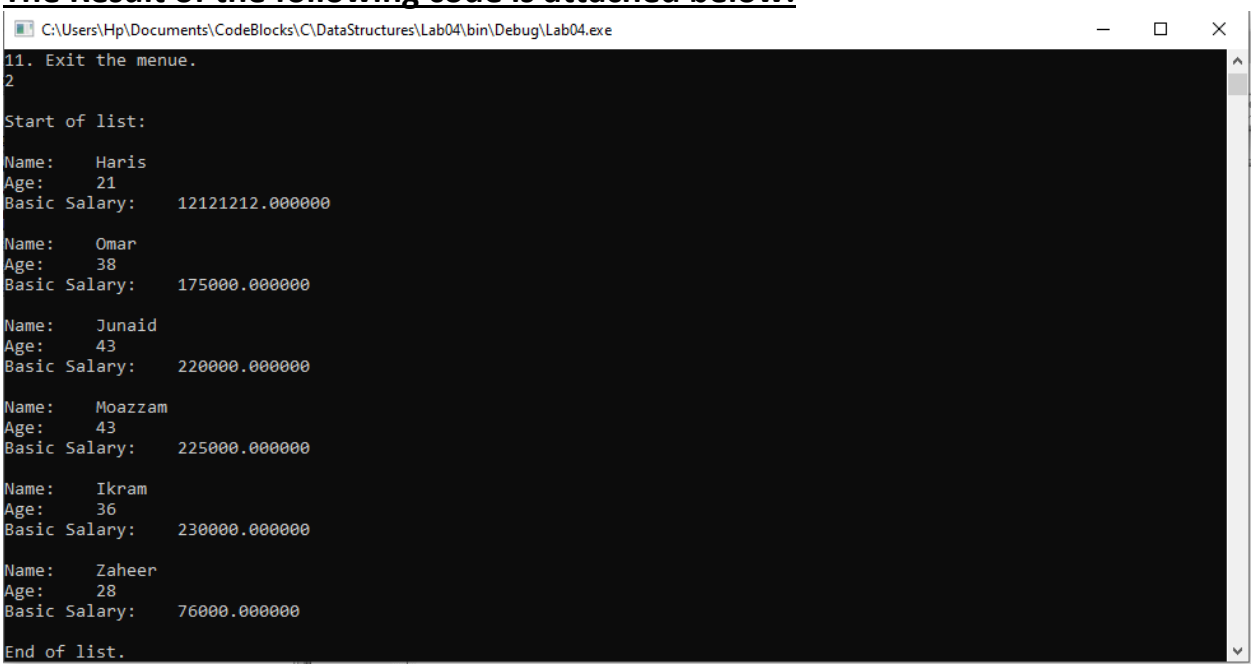


INSERT A NODE AT START

The code is shown below,

```
void  insertNodeAtStart(struct node_d ** head)
{
    /*** To be completed by the students ***/
    if(head==NULL)
 {
 *head= (struct node_d *) malloc(sizeof(struct node_d));

 }

     struct node_d * prevhead=*head;
     struct node_d * temp=*head;

      struct node_d * new_node=(struct node_d *)malloc(sizeof(struct node_d));
      inputNodeData(new_node);

      free(head);
      *head=new_node;
       (*head)->next=temp;
        (*head)->prev=temp->prev;

                while(temp->next!=prevhead)
                {
                    temp=temp->next;
                }

                temp->next=*head;
}
```

The Result of the following code is attached below:

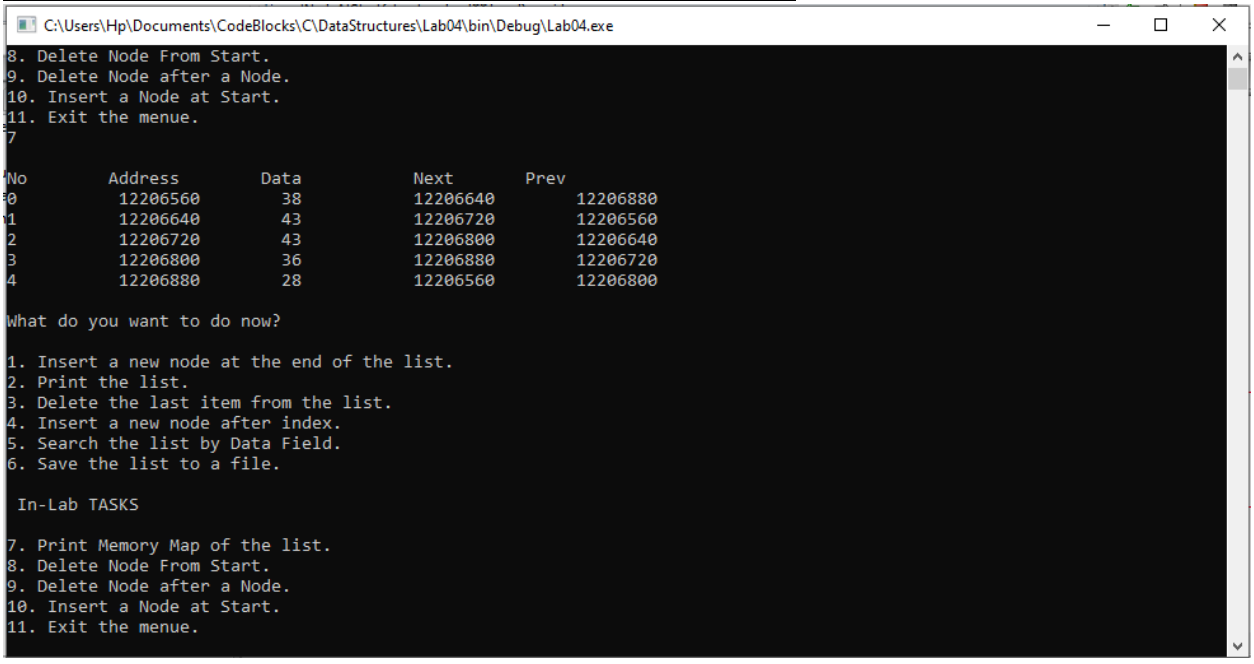


PRINTING MEMORY MAP

The code is shown below,

```
void printMemMap(struct node_d * head)
{
   /*** To be completed by the students ***/
   struct node_d *temp=head;
   struct node_d *fixedhead=head;

   int i=0;
   printf("\nNo        Address        Data           Next       Prev  \n");

   while(temp->next!=fixedhead)
   {

   printf("%d          %u        %d          %u        %u     \n",i,&temp->data,temp->data.age,temp->next,temp->prev);

    temp=temp->next;
    i++;

   }
   printf("%d          %u        %d          %u        %u     \n",i,&temp->data,temp->data.age,temp->next,temp->prev);

}
```

The Result of the following code is attached below:



POST LAB

Question no:3

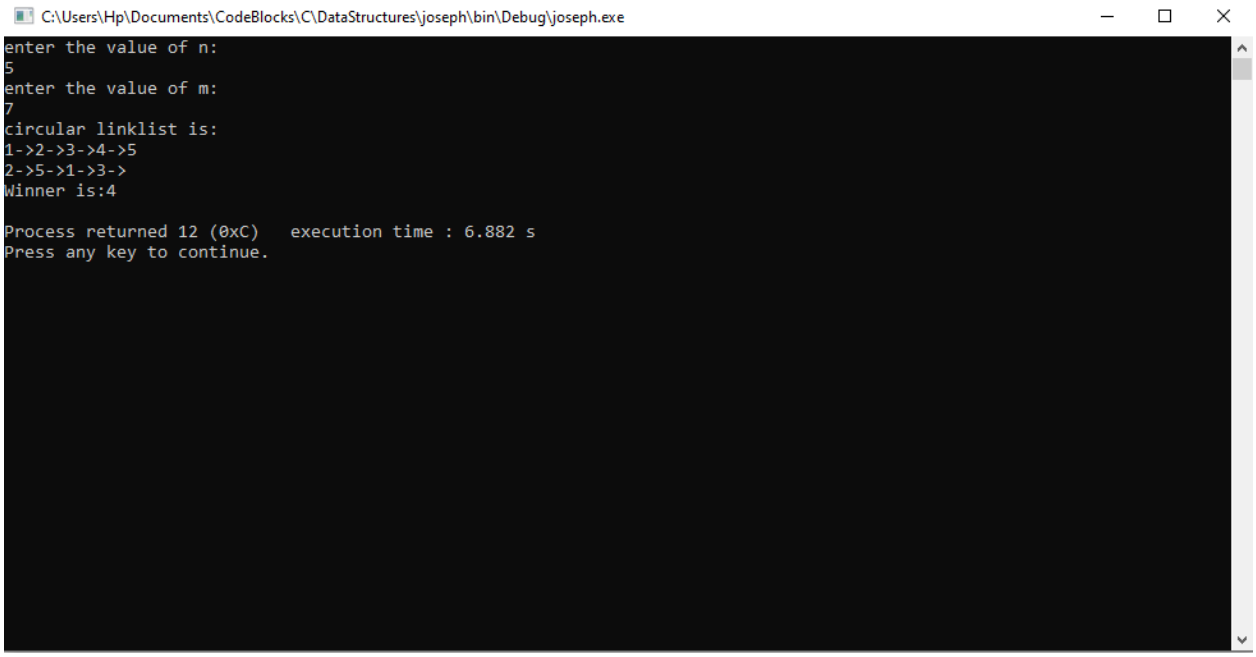
Learn more about at the Josephus Problem from the following links and make a Circular Doubly Linked List simulation of this problem. (Your program should print the remaining people in each iteration).

Solution

The code is shown below for the given program and its results are given below,

```
int josephus(int m,node *front)
{
        node *f;
        int c=1;
        while(front->next!=front)
        {
                c=1;
                while(c!=m)
                {
                        f=front;
                        front=front->next;
                        c++;
                }
                f->next=front->next;
                printf("%d->",front->data); //sequence in which nodes getting deleted
                front=f->next;
        }
        printf("\n");
        printf("Winner is:%d\n",front->data);
        return;
}
```

The Result of the following code is attached below:


C:\Users\Hp\Documents\CodeBlocks\C\DataStructures\joseph\bin\Debug\joseph.exe
enter the value of n:
5
enter the value of m:
7
circular linklist is:
1->2->3->4->5
2->5->1->3->
Winner is:4
Process returned 12 (0xC)   execution time : 6.882 s
Press any key to continue.

THE END