**Assignment 1**

**Name** : Komal Mahadev Potdar

**Roll No**.: 92

**PRN No**.: 12320165

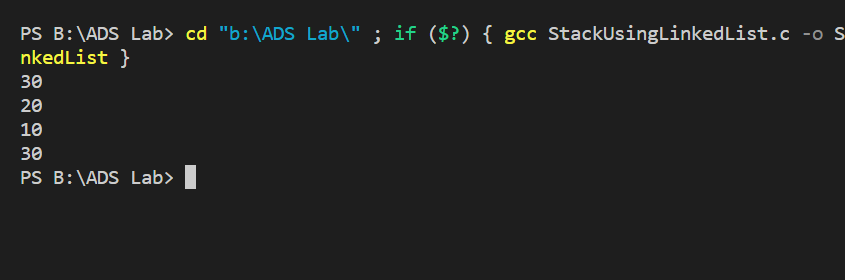
**Div**: CS B SY

**Batch**: 3

1. **Write a program to implement Stack using linked list**

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>  struct node{      int data;      struct node\* next;  }\*top=NULL;  void push(int data1)  {      struct node\* temp = (struct node\*)malloc(sizeof(struct node\*));      temp->data = data1;      temp->next=NULL;      if(top==NULL){          top=temp;      }      else{          temp->next=top;          top=temp;      }  }  void pop()  {      struct node\* temp = top;      if(top!=NULL){          printf("%d",temp->data);          top=top->next;          free(temp);      }      else{          printf("%d",temp->data);          top=top->next;          free(temp);      }  }  void peek()  {      if(top==NULL){          printf("UNDERFLOW\n");          return;      }      printf("%d",top->data);  }  void display()  {      struct node\* temp=top;      if(top==NULL){          printf("EMPTY\n");          return;      }      while(temp!=NULL){          printf("%d\n",temp->data);          temp=temp->next;      }  }  int main( )  {      push(10);      push(20);      push(30);      display();      pop();      return 0;  } |

Output:



1. **Write a program to implement Queue using linked list**

|  |
| --- |
| #include<stdio.h>  #include<stdlib.h>  struct node{      int data;      struct node\* next;  }\*front=NULL, \*rear=NULL;  void enqueue(int data1){      struct node \*temp = (struct node\*)malloc(sizeof(struct node));      temp->data=data1;      temp->next=NULL;      if(front==NULL &&rear==NULL){          front=temp;          rear=temp;      }      else{          rear->next=temp;          rear=temp;      }  }  void dequeue(){      struct node \*temp = front;      if(front==NULL){          printf("Underflow Error\n");      }      else if(front==rear){          temp=front;          printf("Dequeued: %d\n",temp->data);          front=NULL;          rear=NULL;          free(temp);      }      else{            temp=front;          front=front->next;          printf("Dequeued: %d\n",temp->data);          free(temp);      }  }  void traverse(){      struct node\* temp = front;      if(front==NULL){          printf("Empty\n");      }      else{          while(temp!=NULL){              printf("Element %d\n",temp->data);              temp=temp->next;          }      }  }  void peek(){      if(front==NULL)return;      printf("Peek Element %d\n",front->data);    }  void main(){      enqueue(10);      enqueue(20);      enqueue(40);      enqueue(50);      enqueue(60);      traverse();      peek();      dequeue();      dequeue();      dequeue();      traverse();  } |

Output:

