Navlakhi®

Palindrome using Stacks

Methodology and Program

By Abhishek Navlakhi

Semester 3: Data Structures

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
#include<stdlib.h>
#define MAX 100
char stack[MAX];
int top=-1;

void push(char s)
{
 if(top==MAX-1)
 {
  printf("Stack is full.EXITING...\n");
  exit(0);
 }
 else
 {
  top++;
  stack[top]=s;
 }
}


char pop()
{
 char data;
 if(top==-1)
 {
  printf("Stack is empty.EXITING...\n");
  exit(0);
 }
 else
 {
  data=stack[top];
  top--;
 }
 return data;
}
```

```
void main()
{
  char str[50],ans[50];
  int length,i;
  clrscr();
  strset(str,'\0');
  strset(ans,'\0');
  printf("Enter the string:");
  scanf("%s",str);
  length=strlen(str);
  for(i=0;i<length;i++)
  {
    push(str[i]);
  }
  for(i=0;i<length;i++)
  {
    ans[i]=pop();
  }
  printf("Input=%s\n",str);
  printf("Answer=%s\n",ans);
  if(strcmp(str,ans)==0)
    printf("The string is a palindrome!!!");
  else
    printf("The string is not a palindrome.");
  getch();
}
```