1. Write a program to implement push,pop, peek and display operations on stack.

#define MAX 20

#include<stdio.h>

#include<conio.h>

typedef struct stack

{

int data[MAX];

int top

}stack;

void init(stack \*s);

void push(stack \* s,int x);

int pop(stack \* s);

void display(stack \* s);

void peek(stack \* s);

void main()

{stack s;

int ch,n;

init(&s);

do

{

printf("enter choice:\n");

scanf("%d",&ch);

switch(ch)

{

case 1: printf("enter number to be pushed\n");

scanf("%d",&n);

push(&s,n);

break;

case 2:n=pop(&s);

printf("popped element is %d\n",n);

break;

case 3:printf("stack is\n");

display(&s);

break;

case 4:peek(&s);

break;

}

}while(ch<5);

}

void init(stack \*s)

{

s->top=-1;

}

void push(stack \* s,int x)

{

s->top=s->top+1;

s->data[s->top]=x;

}

int pop(stack \* s)

{

int x;

x=s->data[s->top];

s->top=s->top-1;

return x;

}

void display(stack \* s){

int i;

for(i=s->top; i>=0; --i)

{

printf("%d\n", s->data[i]);

}

}

void peek(stack \* s)

{

printf("top element is%d\n",s->data[s->top]);

}

1. Write a program to check well formedness of parentheses.

#include<stdio.h>

#include<conio.h>

int main()

{

char expression[50]; // declaration of char type array

int x=0, i=0; // declaration of two integer type variables

printf("\nEnter an expression");

scanf("%s", &expression);

// Scanning the expression until we reach the end of the expression.

while(expression[i]!= '\0')

{

// Condition to check the symbol is '('

if(expression[i]=='(')

{

x++; // incrementing 'x' variable

}

else if(expression[i]=='{')

{

x++; // incrementing 'x' variable

}

else if(expression[i]=='[')

{

x++; // incrementing 'x' variable

}

// condition to check the symbol is ')'

else if(expression[i]==')')

{

x--; // decrementing 'x' variable

if(x<0)

break;

}

else if(expression[i]=='}')

{

x--; // decrementing 'x' variable

if(x<0)

break;

} else if(expression[i]==']')

{

x--; // decrementing 'x' variable

if(x<0)

break;

}

i++; // incrementing 'i' variable.

}

// Condition to check whether x is equal to 0 or not.

if(x==0)

{

printf("Expression is balanced");

}

else

{

printf("Expression is unbalanced");

}

}

5. Write a program to perform following operations on SLL

1. create a LL(Insert)

2. reverse

3. Display/Traverse