lab 5 work:

#ifndef STACK\_H

#define STACK\_H

template<class KeyType>

class Stack

{ //

public:

// constructor , creates an empty stack

Stack(int max){

maxsize=max;

arr=new KeyType[maxsize];

top=-1;

}

// returns true if Stack is full, otherwise return false

bool IsFull()

{

if(top==maxsize-1)

return true;

else

return false;

}

bool IsEmpty()

{

if(top==-1)

return true;

else

return false;

}

// If Stack is not full, insert item into the Stack. Must be an O(1) operation

void Push(const KeyType item)

{

if(!IsFull() || IsEmpty())

{

if(!arr)

{

arr=new KeyType[this->maxsize];

}

else

{

top++;

arr[top]=item;

}

}

else

{

cout<<"Stack is full"<<endl;

}

}

// If Stack is empty return 0 or NULL;

// else return appropriate item from the Stack. Must be an O(1) operation

KeyType Pop()

{

KeyType a=0;

if(!IsEmpty())

{

KeyType a=arr[top];

top=top-1;

return a;

}

else

{

cout<<"Stack is empty"<<endl;

return 0;

}

}

private:

KeyType \* arr;

int maxsize;

int top;

};

#endif

ccp.

#include "Stack.h"

#include <iostream>

using namespace std;

int main()

{

Stack<int> \*st =new Stack<int>(100);

if(st->IsEmpty())

cout<<"Stack is currently empty"<<endl;

st->Push(1);

st->Push(2);

st->Push(3);

while (!st->IsEmpty())

{

int value=st->Pop();

cout<<value<<endl;

}

system("pause");

return 0;

}