Parbej Alam

Roll no:2300116

Faculty:science and technology(BEIT 3rd sem)

#include <stdio.h>

#define SIZE 5

// Function declarations

void push(int);

void pop();

int peek();

void display();

// Global variables

int stack[SIZE];

int top = -1;

int main() {

int choice, data, r; // 'r' is used to store the peeked value

while (1) {

// Display menu options

printf("\nEnter 1 for push\n");

printf("Enter 2 for pop\n");

printf("Enter 3 for peek\n");

printf("Enter 4 for display\n");

printf("Enter 5 for exit\n");

printf("Your choice: ");

scanf("%d", &choice);

// Menu-driven operations

switch (choice) {

case 1:

printf("Enter data: ");

scanf("%d", &data);

push(data);

break;

case 2:

pop();

break;

case 3:

r = peek();

if (r != -1) { // Check if peek was successful

printf("\n%d is on top\n", r);

}

break;

case 4:

display();

break;

case 5:

printf("Exiting... bye bye\n");

return 0;

default:

printf("Invalid choice. Please try again.\n");

}

}

return 0;

}

// Push operation

void push(int d) {

if (top == SIZE - 1) {

printf("\nStack overflow\n");

} else {

top++;

stack[top] = d;

printf("\nPUSH successful: %d added to stack\n", d);

}

}

// Pop operation

void pop() {

if (top == -1) {

printf("\nStack underflow\n");

} else {

printf("\n%d is popped\n", stack[top]);

top--;

}

}

// Peek operation

int peek() {

if (top == -1) {

printf("\nStack is empty\n");

return -1; // Return -1 to indicate stack is empty

} else {

return stack[top];

}

}

// Display stack elements

void display() {

if (top == -1) {

printf("\nStack is empty\n");

} else {

printf("\nStack elements are:\n");

for (int i = top; i >= 0; i--) { // Print top to bottom

printf("%d\n", stack[i]);

}

}

}