

```

#include<stdio.h>

void push(char element, char stack[], int *top, int stackSize){
    if(*top == -1){
        stack[stackSize - 1] = element;
        *top = stackSize - 1;
    }
    else if(*top == 0){
        printf("The stack is already full. \n");
    }
    else{
        stack[( *top) - 1] = element;
        ( *top)--;
    }
}


void pop(char stack[], int *top, int stackSize){
    if(*top == -1){
        printf("The stack is empty. \n");
    }
    else{
        printf("Element popped: %c \n", stack[( *top)]);

        if(( *top) == stackSize - 1){
            ( *top) = -1;
        }
        else{
            ( *top)++;
        }
    }
}

int main() {
    int stackSize = 4;

```

```
char stack[stackSize];  
int top = -1;  
push('a', stack, &top, stackSize);  
printf("Element on top: %c\n", stack[top]);  
push('b', stack, &top, stackSize);  
printf("Element on top: %c\n", stack[top]);  
pop(stack, &top, stackSize);  
printf("Element on top: %c\n", stack[top]);  
pop(stack, &top, stackSize);  
printf("Top: %d\n", top);  
pop(stack, &top, stackSize);  
return 0;  
}
```

 C:\Users\HP\Documents\stack operation.exe

```
Element on top: a  
Element on top: b  
Element popped: b  
Element on top: a  
Element popped: a  
Top: -1  
The stack is empty.
```

```
-----  
Process exited after 2.071 seconds with return value 0  
Press any key to continue . . .
```