SSN College of Engineering, Kalavakkam
Department of Computer Science and Engineering
III Semester - CSE
UCS 1312 Data Structures Lab Laboratory

Academic Year: 2021-2022 **Batch: 2020-2024**

Date of Assignment: 01.11.2021

Exercise 4: StackADT and its Applications

[CO1,K3]

Create StackADT with the members integer data array, size and top. It contains the following operations

Initialize the stack
void initStack(struct stack *S)
Push an integer element into the stack
void push(struct stack *S,int c)
Pop to remove the top element from the stack by adjusting top
void pop(struct stack *S)
Decrement the top
int top(struct stack *S)
Returns the top element if stack not empty, otherwise -1
int isFull(struct stack *S)
Check whether stack is Empty
int isEmpty(struct stack *S)

Create StackADTImpl.h with the implementations of the above-mentioned operations
Create StackADTAppl.c is menu driven program which utilizes StackADT and StackADTImpl to perform the operations.

1. Demonstrate StackADT with the following test case

init(S,3)
push(S,1)
push(S,2)
push(S,3)
push(S,4) → Stack full
top(S)
top(S)
top(S)
top(S) → Stack empty

2. Write an application to play the following game of two stacks

Alexa has two stacks of non-negative integers, Stack A and Stack B
Alexa challenges Nick to play the following game:
In each move, Nick can remove one integer from the top of either Stack A or Stack B

Nick keeps a running sum of the integers he removes from the two stacks.
Nick is disqualified from the game if, at any point, his running sum becomes greater than some integer

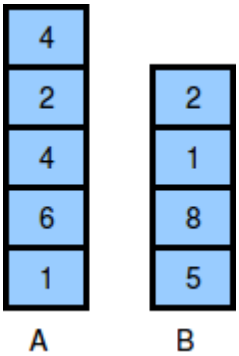
- given at the beginning of the game.
- Nick's *final score* is the total number of integers he has removed from the two stacks.

Given A and B, maxSum, find the score Nick can achieve.

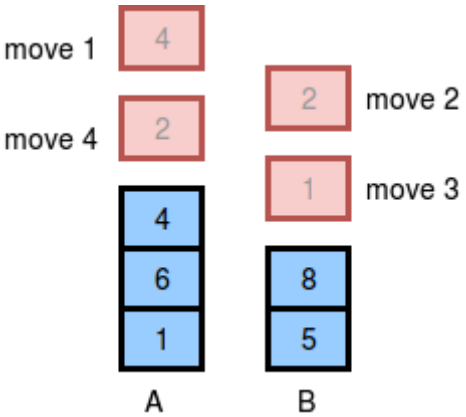
Function Description
Complete the *twoStacks* function in the editor below.
twoStacks has the following parameters:
- *int maxSum:* the maximum allowed sum
- *A:* the first stack
- *B:* the second stack

Returns
- *int:* the maximum number of selections Nick can make

Testcase
Input
The two stacks initially look like this:



The image below depicts the integers Nick should choose to remove from the stacks. Following is on way to remove from the two stacks without the sum exceeding
.



(There can be multiple ways to remove the integers from the stack, the image shows just one of them.)
Output
4