Exercise 8 – Stacks Revisited

Due Friday 16th October 2020 by 23:55. (2 marks)

For this lab we will revisit the heap exercise from Lab 1 but with a few changes.

The first entry in the data file contains the initial stack size
The remaining data will consist of a set of lines containing either:

push *value*

or

pop

with what should be the obvious activity associated with each input – either pushing *value* onto the stack or popping the current top value off the stack.

You should start with a stack array of the specified size.

If you attempt to push data onto a full stack you should

Create a new array of twice the current size,
Copy the data from the old stack to the new stack,
Delete the old stack.

Each time this doubling occurs you should output a line to the console of the form:

If you attempt to pop an empty stack no action should be taken, Do not print an error message.

`Stack doubled from old_size to new_size.`

At the end of the input file you should print a final line of the form:

`Stack contains n entries.`

This is the only output required.

Example:
If the input file looked like:

```
2
push 1
push 1
push 1
push 1
push 1
pop
pop
push1
```

the output would be:

```
Stack doubled from 2 to 4.
Stack doubled from 4 to 8.
Stack contains 5 entries.
```

Note that you do not need to shrink the stack at any time.

As usual, do not use classes or STL.

Submit ex8.*ext* via moodle as usual where *ext* is one of c, cpp, java or py.