Chris Banci

ID: 010031304

January 28, 2015

Stacks #1 - Homework

1a) Composition of stack

|  |
| --- |
| **80** |
| **50** |
| **20** |
| **10** |

S

1a) Sequence of integers popped off the stack:

**40** -> **30**  -> **70**  -> **60**  -> **90**

2) Composition of stack after push and pop operations.

|  |
| --- |
| **R** |
| **T** |
| **P** |
| **C** |

S

String produced from pop operations = “**OMUE**”

3a) Set num to the second element from the top of the stack, leaving the stack without its two elements

**for (i = 1; i > 2; i++) {**

**num = s.pop();  
 }**

3b) set num to the second element from the top of the stack, leaving the stack unchanged.

**x = s.pop()**

**num = s.top();**

**s.push(x);**

3c) Given an integer n, set num to the nth element from the top of the stack, leaving the stack without its top n elements.

**for (i = 1; i < n; i++) {**

**num = s.pop;**

**}**

3d) Given an integer n, set num to the nth element from the top of the stack, leaving the stack unchanged.

**stack t;**

**for (i = 1; i < n; i++) {**

**t.push(s.pop());**

**}**

**num = s.top();**

**for (i = 1; i < n; i++) {**

**s.push(t.pop());**

**}**

3e) set num to to the bottom element of the stack, leaving the stack empty;

**while(!s.isEmpty()){**

**num = s.pop();  
}**

3f) set num to the bottom element of the stack, leaving the stack unchanged.

**stack t;**

**while (!s.isEmpty()){**

**t.push(s.pop);**

**}**

**num = s.top();**

**while (!t.isEmpty()){**

**s.push(t.pop);**

**}**

3g) set num to the nth element from the bottom of the stack leaving the stack unchanged.

**stack t;**

**for (i = SIZE of stack; i > n; i--) {**

**t.push(s.pop());  
}**

**num = s.top();**

**while (!t.isEmpty(){**

**s.push(t.pop();**

4) Describe how the store and exact operations of one-dimensional array can be implemented by using these stack primitive operations on two stacks.

By using two stacks, we can implement store and extract operations of one-dimensional arrays. For example, an implementation of those operations would be as followed:

**stack t;**

**t.push(s.pop());**

**temp = t.pop();**

In this example, we are popping an element from stack s, and pushing it to a new stack called t, effectively extracting and storing the element to variable temp.