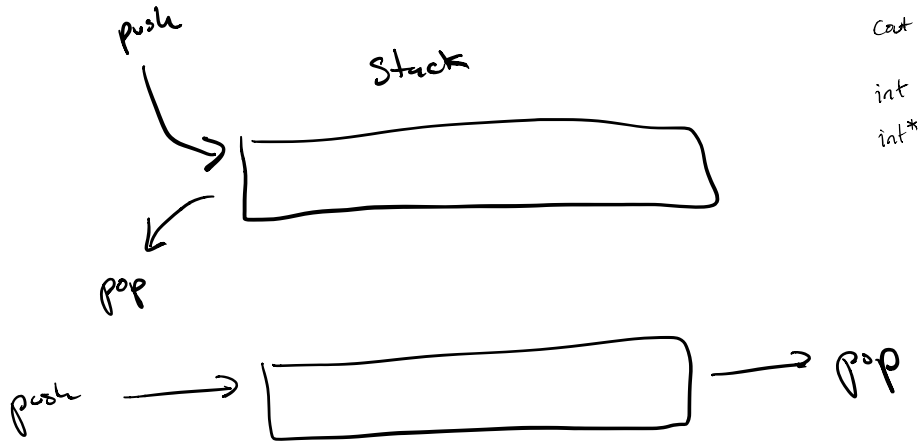
CS 32 - Discussion 1B

Week 4

Stacks, Queues, and some of their applications




```
int* ptr = new int(10);
cout << *ptr;

int val = 10;
int* ptr = &val;
```

Standard library:

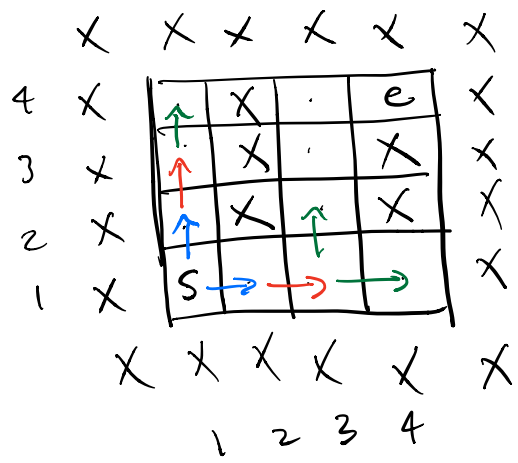
```
#include <stack>
#include <queue>

stack<int> s;
queue<int> q;
```

---

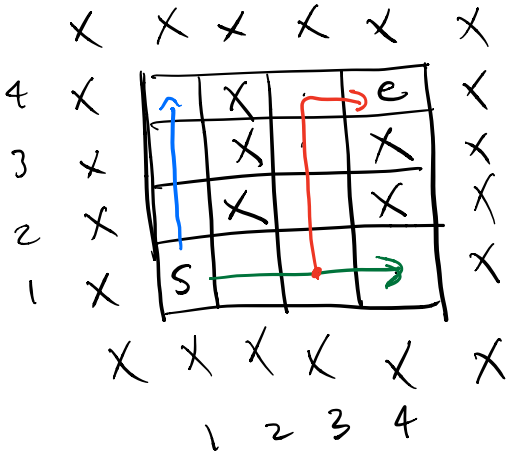
homework 2: problems 1-4

Check if there is a path between two points in a maze.



Breadth-first search

Explore all paths in "parallel"



Depth-first search

Go as far as possible down 1 path before exploring next path

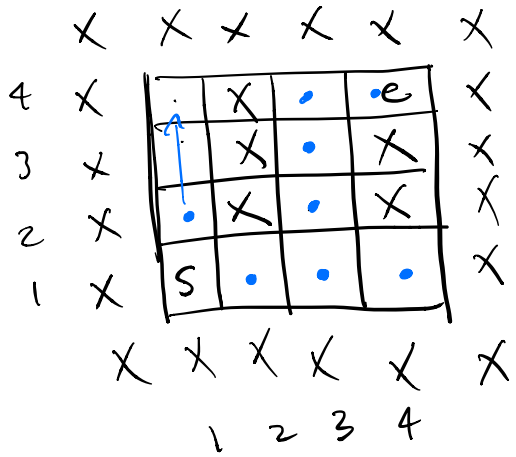
```
search(s, e, maze)
    Stack/queue<Coord> frontier      // contains explored cells at the "edge"
    mark s as 'seen'                 } initialize
    frontier.push(s)

    while (frontier is not empty)    } main loop
        current = frontier.pop()     // returns & removes top value
        if(current == e):
            return true;
        else:
        → for each next_step adjacent to current:
            if next_step is not 'seen':
                frontier.push(next_step)
                mark next_step as 'seen'

    return false;
```

Using stack ⟹ Depth-first search

Using queue ⟹ Breadth-first search



top

frontier: ~~(1,1)~~ ~~(2,1)~~ ~~(3,1)~~ ~~(4,1)~~ ~~(3,2)~~ ~~(3,3)~~ ~~(3,4)~~ ~~(4,4)~~ (1,2)

current = (4,4)