Natasha Porter
Data Structures
Professor Blaer
Fall 2016

Problem 1
printLots(L,P)

```
int sizeOfL = L.size();

for(i=0; i<P.size; i++) {
        if(P.get(i) <sizeOfL) {
                System.out.println(L.get(P.get(i)));
        }
        else {
                System.out.println("There is no element in P with this index.");
        }
break;
}
```

Problem 2

```
int idx1 = 0;
int idx2 = 0;
List L1;
List L2;
List L3;

While( idx1 < L1.size() && idx2 < L2.size()) {
        If(L1.get(idx1) == L2.get(idx2))
                L3.add(get(idx1));
                Idx1++;
                Idx2++;

        Else if(L1.get(idx1) < L2.get(idx2))
                Idx1++;
        Else if(L1.get(idx1) > L2.get(idx2))
                Idx2++;
        System.out.println(L3);
}
```

Problem 3

```
Int topOfStack1 = -1;
Int topOfStack2 = list.size();

        pushS1(x) {
                if(topOfStack2 > topOfStack1) {
                        topOfStack1++;
                        list(topOfStack1) = x;
                }
                else {
                        System.out.println("Stack Overflow");
                }
        }
```

Natasha Porter
Data Structures
Professor Blaer
Fall 2016

```
popS1(x) {
        return topOfStack1;
        topOfStack1--;
}

pushS2(x) {
        if(topOfStack2 > topOfStack1) {
                topOfStack2--;
                list(topOfStack2) = x;
        }
        else {
                System.out.println("Stack Overflow");
        }
}


popS2(x) {
        return topOfStack2;
        topOfStack2--;
}
```

Problem 4
a)
S1: 4,3,2
S2: 8,7,6
S3: 9,5

Train 4 moved to S1.
T3 moved to S1.
T1 moved to output track.
T8 moved to S2.
T2 moved to S1.
T7 moved to S2.
T6 moved to S2.
T9 moved to S3.
T5 moved to S3.

T2 moved to output track.
T3 moved to output track.
T4 moved to output track.
T5 moved to output track.
T6 moved to output track.
T7 moved to output track.
T8 moved to output track.
T9 moved to output track.

b) 1 6 7 8 5 9 4 3 2