APS 105 Lecture 18 Notes

Question 10, Winter 2019

Implement a C function that returns index of 2nd largest # of an array. If array has {3, 9, 7, 5, 9, 8, 2, 4, 9}, return 5, list[5] has 2nd largest integer in array. If all elements are same (ie. no 2nd largest), return -1. Assume all elements are +ve, count > 0. Prototype: int secondLargest(int list[], int count)

Before second largest, I want to find a way to return ind of largest — Practice.

① Identify the problem: return index of largest number

② Start with by example, e.g. small array size with small problem

{3, 9, 7}

i) Start with list[0] as largest

① loop over each element in array

ii) Compare 9 with largest=3, now 9 is largest

iii) Compare 7 with largest=9, 9 remains largest

② Update largest, if list[i] > largest

```
int largest (int list[], int count) {
    int largest = list[0];
    int largestIndx = 0;
    for (int i=1; i<count, i++) {
        if (list[i] > largest) {
            largest = list[i];
            largestIndx = i;
        }
    }
    return largestIndx;
}
```

Back to the 2nd largest question

① Problem: 2nd largest index

② Toy example: {3, [9], 7}

i) Start with largest = list[0] = 3, secondlargest = -1

ii) if list[i] > largest → Update secondlargest with largest
secondlargest = 3, secondlargestInd = 0
→ Update largest & largestInd
9 is largest, 1 largestInd

e.g. i=1
list[1] > largest ⇒
3

opposite

iii) if list[i] < largest, but list[i] > second → Update secondlargest

e.g. i=2; 7<9 7>3
list[2] < largest but list[2] > secondlargest

```
int secondLargest( int list[], int count ){

    int largest = list[0];
    int largestInd = 0;

    int secondLargest = -1;
    int secondLargestInd = -1;

    for (int i=1; i<count; i++){

        if (list[i] > largest){

            secondLargest = list[i];
            secondLargestInd = i;

            largest = list[i];
            largestInd = i;
        }
        else if (list[i] < largest && list[i] > secondLargest)
        {

            secondLargest = list[i];
            secondLargestInd = i;
        }
    }
    return secondlargestInd;
}
```

Question 8 - Winter 2020

An array has 6 integers, write function that prints them in a sending order of their 2nd digit.

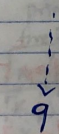
{269, 324, 62, 05, 111, 193}

5, 111, 324, 269, 62, 193

① look for 2nd Digit, if 0 → print in the entire array

② look for 2nd Digit, if 1 → print in the entire array

③



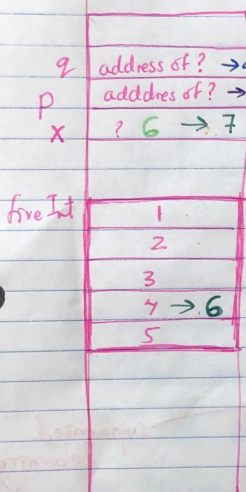
```
bool printedBefore = false;

for (int j=0 ; j <=9 ; j++){
    for (int i=0; i < 6 ; i++) {

        if ((int) ((array[i] % 100)/10) == j ? ){

            if ( printedBefore ){
                printf(", ");
            }
            printedBefore = true;
            printf(" %d", array[i]);
        }

}
```

How to get 2nd dig 7?

537 % 100 = 37 to drop on left %
drop 5

37 / 100 = 3 to drop on right /
drop 7

Fall '18 Q7




```
#include <stdio.h>

int main (void){

    int *p, x;

    int fiveInt [5] = {1, 2, 3, 4, 5};

    int * q;

    p = NULL;  (1)
    q = fiveInt;  (2)
    x = 6;  (3)
    p = &x;  (4)

    printf("A: %d %d \n", x, *p);
                          x = 6
    *(q+3) = *p;
    fiveInt[3]
    *p = *q + *(q+3);
    x = fiveInt[0] + fiveInt[3]
    printf("B: %d %d %d", x, *p, *q);
    return 0;
}
```

A: 6 6
B: 7 7 1