**페이스북 인턴(박사)인터뷰**

-**평가방법**: 인터뷰어와 화면을 공유하면서 동시에 전화통화 진행, 메모장에 코딩을 하면 인터뷰어가 돈다/안 돈다 평가를 내려줌. 만약 돌지 않을 시 스스로 에러 찾아내면 괜찮은 평가를 받을 수 있지만 아니면 탈락.

-**평가기준**: 얼마나 막힘없이 영어로 본인의 아이디어를 설명할 수 있는가.

C++을 이용하여 본인의 아이디어를 단번에 구현할 수 있는가.  
🡪 문제를 잘 풀어도 이 기준에 미달되면 탈락

**-문제**

**Question 1)** Sample an index of a maximal number in an array, with a probability of 1/(number of maximal numbers)

Requirements) Solve in **15 minutes** / O(n) time, O(1) space / C++, using vector

**Answer)**

int find\_maxIndex(vector<int> a){

int count = 1;

int maxElement = a[0];

for(int i = 1; i < a.size(); i++){

if(a[i] == maxElement){

count ++;

} else if(a[i] > maxElement){

count = 1;

maxElement = a[i];

}

}

int occurrence = rand() % count + 1;

int occur = 0;

for(int i = 0; i < a.size(); i++){

if(a[i] == maxElement){

occur++;

if(occur == occurrence) return i;

}

}

}

**Question 2)** Find longest subarray that has elements in increasing order?

Requirements) Solve in **15 minutes** / O(n) time, O(1) space / C++

**Answer)**

void find\_longestSub(int a[]){

int max = 1;

int start = 0;

int aux = 0;

int count = 1;

for (int i = 1; i < sizeof(a)/sizeof(a[0]); i++) {

if (a[i] > a[i - 1]) {

count++;

} else {

aux = i;

count = 1;

}

if (count > max) {

max = count;

start = aux;

}

}

cout<< “Length of LongestSub: ” << max << “, index: (” << start << “, ” << (start+max-1) << “)”

}