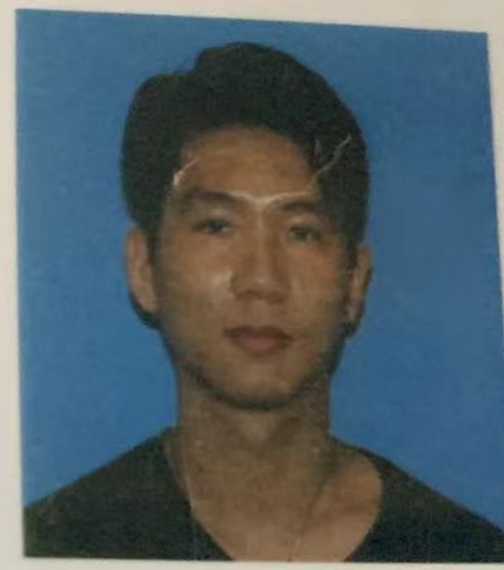
QUEENS
COLLEGE

Q Card



STUDENT

RUIXUAN ZHANG

23663661

1.

2. To get ~~largest~~ largest $O(\text{sqrt}(n))$ number
$k = O(\text{sqrt}(n))$
Total time = kth samllest + sorting
$= O(n) + O(\text{sqrt}(n) * \log(\text{sqrt}(n))) = O(n)$

3. a) optimal value for each and every node has j contain j > 1 where the node contain i.

opt(i) = max(opt(j)+1)

Out[i] = longest ~~length~~ path between V1 to Vi

Out[i], Max(out[i]+v)+1 while v < 1

```
int longestPath() {
  for (int i=1; i<=v; --i){
    out[i]=-1;
    Out[i]=0;
      for(int i=1; i<=v, i++){
      if(out[i] >=0) {
          for(i in 'v' to V){
          out[v]-Max(out[v]+1, out[v]);}}
            return out[v];}
```

b) - Two nodes can be connected such that V1 is connected to $(V_2, V_3 \ldots V_n)$

- V_j is connected with $(V(i+1), (i+2), \ldots V_n)$
- number of vertices as $V=n$

  $edges = E = n \times (n+1)/2 = O(V^2)$

$Opti[i] = Max(1 <= k < i) \in E(Opti[k]+1)$

- $P[1]=0$
- for $i=2$ to n
- Algorithm see for all edges going into i store all max longest path has edge +1
- It set $P[i]$ to this value

  $\boxed{\text{run time} = O(n^2)}$
- it the loop complete $P[n]$

4. One bowl = 40 = 1h and 4lbs
One mug = 50 = 2h and 3lbs
Total = 15

	Bowls	Mugs
No #	24	8
Profit	40	50
labor	1	2
clay	4	3

~~Maximize Prof~~ maximum Profit is 136.

5. To find longest Palindromic subsequence, we have fine string that given and reverse the given string.

```
int dp[1001][1001];
int lcs(string s, string t, int n, int m){
    if(n==0||m==0) return 0;
    if(dp[n][m]!=-1) return dp[n][m];
    if(s[n-1]==t[m-1]) return dp[n][m]=1+lcs
        (s,t,n-1,m-1);
    else return dp[n][m]=max(lcs(s,t,n-1,m),
        ~~lcs~~ lcs(s,t,n,m-1));
int main(){
    int n; cin>>n
    string s; cin>>s;
    string t=s;
    reverse(t.begin(), t.end());
    memset(dp, -1, sizeof(dp));
    cout<< lcs(s,t,n,n);
}
```

6 Part 1) - Every Problem in NP can be reduced to the said Problem in polynomial time
- It is in NP.

Part 2) The maximum ~~i~~ independent set Problem of a graph can be reduced to the largest complete subgraph

1. Find all the independent vertices in the graph.
2. Take set C be the complement graph of MIS. It has the vertices that are not in MIS and has edges that are not ~~inc~~ ~~incid~~ incident of any of the vertices in MIS.
3. If number of vertices in C is >=k, answer is ~~e~~ Yes, else no.

Complete subgraph of size = n - |MIS|.