Backtracking

int magicfun (int N) } if (N = -0) return D else return magic fun (N/2) + 10 + (N/2) /N×10+(71.2) = 111 magicfun (7). 1 120 + (3/2) = 11 magictur(3) = 0710 + (1/.2) = 1 magictum(3) magisfun(1) magicfun (7)

TC = OC (USN)

SCZ Ollegn)

fun ("SCROLL",0)

Quiz 2

magicfun(0) -

void fun (char 21), int x) {

print(s)

char temp

if (x < s.length/2) {

temp = s(x)

Subarray us Subset

arr[] = \ \ 1,2,33

Subarrays: (1) [2]
$$(2,3)$$
 (1) [2] $(1,2,3)$ (1) $(2,3)$ (1) $(2,3)$ (1) $(2,3)$ (1) $(2,3)$ (1) $(2,3)$ (1)

Buntion

leinen an array. Print all of its subsets.

A= [1,2,3]

A(N) => 2N subsets

olp: 33

313

91,23

31,37

{2,3}

323

433

31,2,33

A=[1,2,3,4]

3 3

۶ I 3

91,23

31,37

{2,3}

323

433

3 43

81/43

91,2,43

31,3,43

{2,3,4}

32,47

43,43

$$\{1,2,3\}$$
 $\{1,2,3\}$
 $\{1,2,3\}$
 $\{1,2,3\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0\}$
 $\{3,0$

total calls 2
$$2^{\circ} + 2^{1} + 2^{2} + 2^{3} + \dots + 2^{N}$$

$$= 2^{N+1} - 1 = 0(2^{N})$$

$$= 4 \cdot 2^{N} \neq 1$$

```
subsets (a, 0, 83)
void subsers (int al), int i, list(int) currect) }
       if ( i = = a.size) }
           print (currect)
       11 2 option
       41. Pick a(i) in the subset
           currect add (ali)
            subsets (a, i+1, current)
            currsc4. remove last() -> property of back+racking
       112. Don't Pick au)
            subschs (a, i+1, current)
                                 TC=0(2N)
```

SC=O(N')

```
(10,70,30)
   void subsets (int all, int i, list (int) correct) }
         + ( i = = a.size) {
               print (currect)
               return
          curred add (att)) (10)
          subsets (a, ix, current)
           wrosed remove last()
          subsets (a, iet, current)
                                        (10)
              (10,20,30)
void subsels (int all), but i, list sid > curreset) }
     if ( i = = a.size) $
           brint (currect)
           return
                                                                                                          (10)
                                                                                (10,20,30) 2
      curred add (ali) (10,20)
                                                                 void subsets (int all, int i, list (int) comment) }
      subsets (a, it, current)
                                                                       if ( i = = a.size) }
       wrsc. remove last ()
                                                    9
                                                                             print (currect)
      subsch ( a, it, current)
                                                                             neturn
                                                                        curred add (all) (10,30)
                                        (10/20)
               10,20,30) 2
                                                                        subsets (a, i+1, current)
void subsets ( int all , int i', list (int ) correct ) }
                                                                         currsc4. remove last()
    If ( i = = a.size) $
                                                                         subsch (a, i+1, current)
           print (currect)
           return
                                                                                                           (10,20)
      curred add (au) (10,20,30)
      subsets (a, it, current)
                                                                                             prost (wreset) -> (10,50)
       wrsc removelast (10,20)
                                                                                            urred add (all)
      subsets (a, it, current)
                                                                                           subsets (a, in, current)
subsets (a, in, current)
subsets (a, in), current)
                                       (10,20,30)5
                                                                                                 (10,20)
              (10,20,30) 3
                                                        void subsets (int all), int i', list (int) current) }
void subsets ( int at), int i, list(int) currect) }
                                                               4 ( i = = a.size) $
      If ( i = = a.size) $
           print (currect) - (10,20,30)
                                                                    print ( currect) -> (10,20)
                                                                   Morn
           Morn
                                                               currect add (ali)
      currect add (ali)
                                                               subsets (a, i+1, current)
      subsets (a, i+1, current)
                                                               wrosed. remove last()
       wrose. remove last()
                                                               subsets (a, iel, current)
       subscho (a, it, current)
```

Question Cineu a string with distinct characters, print all permutation. abc cate acc cba bac bac

String of stru
$$N \rightarrow N$$
] permutation.

$$[a-1], [a, b] \stackrel{?}{=} [a-1], [a, b-1], [a, b-1$$

```
Ttype bool
void permutations (str, i, aus, vis) }
     if (i == str. size()) } permutations ("abe", 0,
                                        [---], [0,0,0])
        print (aus)
          return
     for ( j=0 to n-1) { // n= vis.size()
         if ( vis(j) = = false ) }
             anslil = strlj]
            vis(j) = Four
             permutations (str, it, aus, vis)
          3 anslit 2 10 not needed
   TL > total func. calls X TL of one func. call
                  NI & DIN)
       TC 2 (NxN!) => O((0+1)!)
            SC = O(N)
```

