

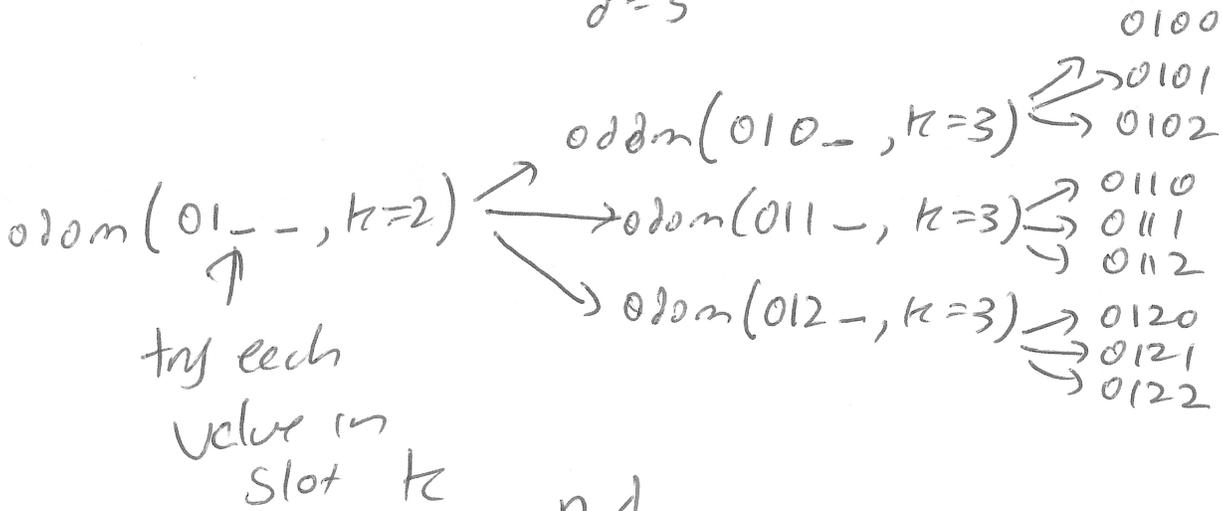
0	1	2	3
0	1	2	2

$n=4, d=3$
(0,1,2)

slots filled in
index of current slot to fill

0	0	0	0
0	0	0	1
0	0	0	2
0	0	1	0
			⋮
2	2	2	2

odom ([0][][][], $k=2, n=4, d=3$)



odom (arr, k) {

if (k == n) {
print (arr)
return;

}
for (int x = possible) {
arr[k] = x
odom (arr, k+1, n, d)

}
}

Combinations

$\{ \overset{8}{a}, \overset{4}{b}, \overset{2}{c}, \overset{1}{d} \}$

$\{b, c, d\}$

$\{ \}$	0	$\{b\}$	4
$\{d\}$	1	$\{b, d\}$	5
$\{c\}$	2	$\{b, c\}$	6
$\{c, d\}$	3	$\{b, c, d\}$	7

$\{a\}$	8	$\{a, b\}$	12
$\{a, d\}$	9	$\{a, b, d\}$	13
$\{a, c\}$	10	$\{a, b, c\}$	14
$\{a, c, d\}$	11	$\{a, b, c, d\}$	15

0 0 0 0	$\{ \}$
0 0 0 1	$\{d\}$
0 0 1 0	$\{c\}$
0 0 1 1	$\{c, d\}$

ALT SOL \nearrow

- 1 4 5 6
- 1 4 5 7
- 1 4 5 8
- 1 4 5 9
- 1 4 5 10

combo (1 | 4 | 5 | , $k=3$, $n=10$, $tot=4$)

Permutation

all orderings of distinct items
 $0, 1, 2, \dots, n-1$

$n=4$

0 1 2 3

0 1 3 2

0 2 1 3

⋮

3 2 1 0

perm

		2				
3	0	1				

used

T	F	F	F	F	F	F
---	---	---	---	---	---	---

 T
 F

ODOM

```
for (int i=0; i<n; i++)
```

```
arr[k] = i
```

BUT

```
for (int i=0; i<n; i++) {
```

```
if (used[i]) continue;
```

```
arr[k] = i
```

```
used[i] = 1;
```

```
perm(arr, k+1, n);
```

```
used[i] = 0;
```

```
}
```

Upword

$k=1$

$n=3$

min
 char
 skip

length

ACE

ADF

AEG

ACF

ADG

⋮

ACZ

ADZ

AEZ

skA

upword (

		↓		
a	c			

 , $\underline{k=2}$, $\underline{skip=1}$) arr[k-1] + skip+1

slot