Python

```python
# -*- encoding:utf-8 -*-
import copy

def combination(elem):
    '''         '''
    for i,v in enumerate(elem):
        for v2 in elem[i+1:]:
            print (v, v2)

def permutation(slot,elem, size):
    '''         '''
    if len(elem) == 1:
        slot[size - 1] = elem[0]
        print slot
    else:
        for i,v in enumerate(elem):
            slot[size - len(elem)] = v
            e2 = copy.copy(elem)
            e2.remove(v)
            permutation(slot, e2, size)
    return

if __name__ == '__main__':
    elem = range(4)
    slot = [-1 for x in elem]

    combination(elem)
    permutation(slot, elem, len(elem))
```

```
(0, 1)
(0, 2)
(0, 3)
(1, 2)
(1, 3)
(2, 3)
[0, 1, 2, 3]
[0, 1, 3, 2]
[0, 2, 1, 3]
[0, 2, 3, 1]
[0, 3, 1, 2]
[0, 3, 2, 1]
[1, 0, 2, 3]
[1, 0, 3, 2]
[1, 2, 0, 3]
[1, 2, 3, 0]
[1, 3, 0, 2]
[1, 3, 2, 0]
[2, 0, 1, 3]
[2, 0, 3, 1]
[2, 1, 0, 3]
[2, 1, 3, 0]
[2, 3, 0, 1]
[2, 3, 1, 0]
[3, 0, 1, 2]
[3, 0, 2, 1]
[3, 1, 0, 2]
[3, 1, 2, 0]
[3, 2, 0, 1]
[3, 2, 1, 0]
```

```
def combi(n , r):
    p = 1
    for i in range(1, r+1):
        p = p * (n - i + 1) / i
    return p

if __name__ == "__main__":
    print combi(4, 2)
```

6

1

Pytho3.x   functiools   import   2.7x

```
>>> from functools import reduce
>>> cmb = lambda n,r:reduce(lambda x,y:x*y,range(n,n-r,-1))/reduce(lambda x,y:x*y,range(r,1,-1))
>>> cmb(7,3)
35.0
```