

CM1103 PROBLEM SOLVING WITH PYTHON

COUNTING

Dr Jing Wu

School of Computer Science & Informatics

Cardiff University

Example

How many ways are there that two prize winners can be chosen from the set { Andrew, Bob, Clare, Deb}?

Example

How many outcomes are possible from rolling two dice?
Does it matter what colour they are?

Overview

- Rules for counting
 - The product rule
 - The sum rule
- What are we counting
 - Permutations
 - Combinations

Example

Suppose that a printer can print in four typefaces: Pica, Elite, Courier, and Times Roman. Suppose also that each typeface can appear in one of three styles: normal, bold, and italic. A font is a combination of a particular typeface with a particular style, such as Courier Italic. How many fonts are there?

The product rule

If an operation consists of k steps, where:

- 1st step can be performed in n_1 ways;
- 2nd step can be performed in n_2 ways
regardless of how 1st step was performed;
- ...
- k^{th} step can be performed in n_k ways
regardless of how preceding steps were performed;

Then the entire operation can be performed in $n_1 \times n_2 \times \cdots \times n_k$ ways.

Example

Example: the number of elements in a Cartesian product

Suppose A_1, A_2, A_3 are sets with n_1, n_2, n_3 elements, respectively. How many elements are there in the set $A_1 \times A_2 \times A_3$?

Example

Example: two digit numbers

How many integer numbers (in decimal) can be represented using two digits?

$$a_1 a_2$$



For the same problem, if a_1, a_2 subject to a constraint: $a_1 + a_2 \leq 15$, can we still apply product rule?

Example

Example: bank pins

If a bank pin consists of 4 digits (a digit is a number from 0 to 9), how many possible pins are there?

Example: bank pins

If a bank pin consists of 4 **different** digits, how many possible pins are there?

Example

Example: nested loops

```
-----  
for i in range(0,4):  
    for j in range(0,3):  
        # Do some work  
-----
```

How many times will *some work* be executed?

The sum rule

If $\{S_1, S_2, \dots, S_k\}$ is a *partition* (see Sets handout) of a finite set S , then:

$$|S| = |S_1| + |S_2| + \dots + |S_k|$$

Example

Example: password

Suppose we insist users choose a password of exactly 4 characters with a pattern: numbers followed by letters. There should be at least one number and at least one letter (only numbers & letters allowed). How many possible passwords are there?



What if there's no restriction on order?

Example

Example: two digit numbers

How many integer numbers (in decimal) can be represented using two digits?
 $a_1 a_2$, if a_1, a_2 subject to a constraint: $a_1 + a_2 \leq 15$.

Permutations

Definition (permutation)

A permutation of a set of objects is an order of the objects in sequence.

Example:

$$S = \{a, b, c\}$$

Permutations of S :

Counting permutations

A set A of n objects has $n! = n \times (n - 1) \times (n - 2) \times \cdots \times 1$ permutations.

Why?

We have:

n choices from A for the 1st element p_1 of the permutation

$n - 1$ choices from $A - \{p_1\}$ for the 2nd element p_2 of the permutation

$n - 2$ choices from $A - \{p_1, p_2\}$ for the 3rd element p_3 of the permutation

... ..

1 choice for the last

so follows by product rule

Example

Example: permutations of the letters in a word

How many ways can the letters in the word *COMPUTER* be arranged in a row?

Example: permutations of the letters in a word

How many ways can the letters in the word *COMPUTER* be arranged if the letters *CO* must remain next to each other (in order) as a unit?

r -permutations

Definition (r -permutation)

An r -permutation of a set of n elements is an ordered selection of r elements taken from the set without repetition.

Definition ($P(n, r)$)

The number of r -permutations of a set of n elements is given by:

$$P(n, r) = \frac{n!}{(n - r)!}$$

Example

Suppose we have a squad of 14 cricketers. How many batting line-ups of 11 players are there?

r -permutations

Why $P(n, r) = \frac{n!}{(n-r)!}$?

To calculate $P(n, r)$ for a set A we have:

n choices from A for the 1st element p_1 of the r -permutation

$n - 1$ choices from $A - \{p_1\}$ for the 2nd element p_2 of the r -permutation

$n - 2$ choices from $A - \{p_1, p_2\}$ for the 3rd element p_3 of the r -permutation

... ..

$n - (r - 1)$ choices for the r^{th} element of the r -permutation

So, follows by product rule:

$$\begin{aligned} P(n, r) &= n \times (n - 1) \times (n - 2) \times \cdots \times (n - (r - 1)) \\ &= \frac{n \times (n - 1) \times (n - 2) \times \cdots \times (n - (r - 1)) \times (n - r) \times \cdots \times 1}{(n - r) \times \cdots \times 1} \\ &= \frac{n!}{(n - r)!} \end{aligned}$$

Permutations in Python

```
>>> import itertools
>>> for perm in itertools.permutations("abcd"):
...     print(perm)
```

```
>>> len(list(itertools.permutations("123")))
6
>>> list(itertools.permutations("123"))
[('1', '2', '3'), ('1', '3', '2'), ('2', '1', '3'), ('2', '3', '1'), ('3', '1', '2'), ('3', '2', '1')]
```

```
>>> for perm in itertools.permutations("abcd", 2):
...     print(perm)
```

r -combinations

Definition (r -combination)

An r -combination of a set of size n is a subset of size r

Definition ($C(n, r)$)

Let $C(n, r)$ denote the number of r -combinations of a set of size n . Then:

$$C(n, r) = \frac{n!}{r!(n-r)!}$$

Why?

An r -permutation consists of two steps:

| | |
|--|-----------|
| selection of r elements to form a subset of size r | $C(n, r)$ |
| order the r elements into a sequence | $P(r, r)$ |

Apply product rule: $C(n, r) \times P(r, r) = P(n, r)$

So $C(n, r) = \frac{P(n, r)}{P(r, r)} = \frac{n!}{r!(n-r)!}$

Example

How many different ways are there of choosing a team of 11 football players from a squad of 23?

What if we require a goalkeeper to be chosen, and there are 3 in the squad?

Example

Example: password

Suppose we insist users choose a password of exactly 4 characters. There should be at least one number and at least one letter (only numbers & letters allowed). How many possible passwords are there?

The password example, but with no restriction on order of numbers/letters.

Combinatorial equivalence

Some counting problems can be made easier by recognising when there is a one-to-one correspondence to another problem

Example

| | |
|--|--|
| How many ways are there of choosing 2 elements from the set $\{1, 2, \dots, 9\}$? | |
| How many ways are there of choosing 7 elements from the set $\{1, 2, \dots, 9\}$? | |

$$C(n, r) = C(n, (n - r))$$

r-combinations in Python

```
>>> for comb in itertools.combinations(range(1,5),3):  
...     print(comb)
```

Repetition

Theorem (Permutations with repetition)

The number of r -permutations of a set of size n is n^r if repetition is allowed.

Apply product rule

Theorem (Combinations with repetition)

The number of r -combinations of a set of size n is $C(n + r - 1, r)$ if repetition is allowed.

Example:

$$S = \{a, b, c\}$$

2-combinations of S (repetition allowed):

Why $C(n + r - 1, r)$?

| a | | b | | c | |
|-----|--|-----|--|-----|------------|
| | | | | | |
| x | | x | | | $\{a, b\}$ |
| x | | | | x | $\{a, c\}$ |
| | | x | | x | $\{b, c\}$ |
| x x | | | | | $\{a, a\}$ |
| | | x x | | | $\{b, b\}$ |
| | | | | x x | $\{c, c\}$ |

Draw lines between objects
 $n - 1$ lines

Put a cross below the objects
which are selected

r crosses

In total, there are $n - 1 + r$
lines and crosses

We regard them as $n + r - 1$
positions

The selection with repetition problem can be translated as:
Among the $n + r - 1$ positions, choose r for crosses.

$$C(n + r - 1, r)$$

Example

How many different strings of length 5 can be formed using only consonants?

A doughnut shop sells 4 different types of doughnut. How many different ways are there of choosing 6 doughnuts?

Indistinguishable objects

Theorem (Permutations with indistinguishable objects)

Suppose we have n_1 objects of *type* 1, n_2 of *type* 2, ..., n_k of *type* k with $n = n_1 + \cdots + n_k$. The number of permutations of these objects is:

$$\frac{n!}{n_1! n_2! \cdots n_k!}$$

Example

How many strings can be made by rearranging the letters of **scarceness**?

Why $\frac{n!}{n_1!n_2!\cdots n_k!}$?

n_1 objects of *type 1*, n_2 objects of *type 2*, ..., n_k objects of *type k*

$$n = n_1 + n_2 + \cdots + n_k$$

 n positions

Step 1: from the n positions, select n_1 for *type 1* $C(n, n_1)$

Step 2: from the remaining $n - n_1$ positions, select n_2 for *type 2*

... $C(n - n_1, n_2)$

Step k: from the remaining $n - n_1 - \cdots - n_{k-1} = n_k$ positions, select n_k for *type k* $C(n - n_1 - \cdots - n_{k-1}, n_k)$

Follow product rule:

$$\begin{aligned} & C(n, n_1) \times C(n - n_1, n_2) \times \cdots \times C(n - n_1 - \cdots - n_{k-1}, n_k) \\ &= \frac{n!}{n_1!(n-n_1)!} \times \frac{(n-n_1)!}{n_2!(n-n_1-n_2)!} \times \frac{(n-n_1-n_2)!}{n_3!(n-n_1-n_2-n_3)!} \times \cdots \times \frac{(n-n_1-\cdots-n_{k-1})!}{n_k!0!} \\ &= \frac{n!}{n_1!n_2!\cdots n_k!} \end{aligned}$$

Summary

You should be able to:

- Count and list the permutations, r -permutations, and r -combinations of sets by hand, by formula and using Python
- Count permutations and combinations when repetition is allowed or there are indistinguishable objects in the set
- Be able to solve counting problem by using the product and sum rules to divide and combine answers

Definitions covered

- Permutation, r -permutation, r -combination
- $P(n, r)$, $C(n, r)$