Lab 8 - Counting and The Pigeonhole Principle
Computer Science 1FC3

Author: Dai Tri Man Lê
ledt@mcmaster.ca

1 The Basics of Counting

Example 1: (Product Rule)

There are three available flights from Indianapolis to St. Louis and, regardless of which of these flights is taken, there are five available flights from St. Louis to Dallas. In how many ways can a person fly from Indianapolis to St. Louis to Dallas?

Solution. Since there are three ways to make the first part of the trip and five ways to continue on with the second part of the trip, regardless of which flight was taken for the first leg of the trip, by the *product rule* there are $3 \cdot 5 = 15$ ways to make the entire trip. **(End of Solution.)**

Example 2: (Generalized Product Rule)

A certain type of push-button door lock requires you to enter a code before the lock will open. The lock has five buttons, numbered $1, 2, 3, 4, 5$.

1. If you must choose an entry code that consists of a sequence of four digits, with repeated numbers allowed, how many entry codes are possible?

2. If you must choose an entry code that consists of a sequence of four digits, with no repeated digits allowed, how many entry codes are possible?

Solution.

1. We need to fill in the four blanks in $\sqcup \sqcup \sqcup\sqcup$, and each blank can be filled in with any of the five digits 1,2,3,4,5. By the generalized product rule this can be done in $5 \cdot 5 \cdot 5 \cdot 5 = 5^4 = 625$ ways.

2. We need to fill in the four blanks in $\sqcup \sqcup \sqcup\sqcup$, but each blank must be filled in with a distinct integer from 1 to 5. By the generalized product rule that can be done in $5 \cdot 4 \cdot 3 \cdot 2 = 120$ ways. **(End of Solution.)**

2 Problem Set 1

1. Count the number of print statements in this algorithm:

```
for i:=1 to n
begin
   for j:=1 to n
       print hello
   for k:=1 to n
       print hello
end
```

2. Count the number of print statements in this algorithm:

```
for i:=1 to n
 begin
  for j:=1 to i
    print hello
  for k:= i+1 to n
    print hello
 end
```

Remark. The counting techniques using in 1. and 2. are exactly what we used when analyzing the complexity of a program with loops. **(End of Remark.)**

3. How many numbers with two digits are there in $\mathbb{N}$?

4. How many numbers with two *distinct* digits are there in $\mathbb{N}$?

5. Find the number of strings of length 10 of letters of the alphabet with no repeated letters

   (a) that contain no vowels.

   (b) that begin with a vowel.

   (c) that have C and V at the ends (in either order).

   (d) that have vowels in the first two positions.

6. Ten men and ten women are to stand in a row.

   (a) Find the number of possible rows.

   (b) Find the number of possible rows if no two people of the same sex stand adjacent.

7. Find the number of integers from 1 to 400 inclusive that are:

   (a) divisible by 6.

   (b) not divisible by 6.

   (c) divisible by 6 and 8.

   (d) divisible by 6 or 8.

3 The Pigeonhole Principle

Theorem 1: (The Pigeonhole Principle)

If $k+1$ objects are placed into k boxes, then there is at least one box containing two or more of the objects.

Example 1:

There are 13 people at a house party. Prove that at least two of them must have birthdays in the same month.

Solution. We have 13 people but only 12 months, so at least two of them must have birthdays in the same month. **(End of Solution.)**

Theorem 2: (The Generalized Pigeonhole Principle)

If N objects are placed into k boxes, then by the Pigeonhole Principle there is at least one box containing at least $\lceil N/k \rceil$ objects.

Remark. The Generalized Pigeonhole Principle works for the case when the number of objects N exceeds a multiple of the number of boxes k. **(End of Remark.)**

Example 2:

There are 37 people at a house party. Prove that at least four of them must have birthdays in the same month.

Solution. We have 37 people but only 12 months, so by the Generalized Pigeonhole Principle at least $\lceil 37/12 \rceil = 4$ of them must have birthdays in the same month. **(End of Solution.)**

4 Problem Set 2

1. Prove that in any group of three positive integers, there are at least two whose sum is even.
2. If positive integers are chosen at random, what is the minimum number you must have in order to guarantee that two of the chosen numbers are congruent modulo 6.
3. Prove that in any set of 700 English words, there must be at least two that begin with the same pair of letters (in the same order), for example, ST OP and ST ANDARD.
4. Each type of machine part made in a factory is stamped with a code of the form letter-digit-digit, where the digits can be repeated. Prove that if 8000 parts are made, then at least four must have the same code stamped on them.
5. Each student is classified as a member of one of the following classes: Freshman, Sophomore, Junior, Senior. Find the minimum number of students who must be chosen in order to guarantee that at least eight belong to the same class.

5 Some Maple

Maple provides you two packages which deal with combinatorial structures **combinat** and **combstruct**. In this lab, we will use only the package **combstruct**. To know more about these two packages, type

```
?combinat
?combstruct
```

To access the services provided by the **combstruct** package, type

```
with(combstruct);
```

Use can use function **allstructs** to generate all the objects of a given size

```
allstructs(Combination([apple,orange,pear]), size=2);
allstructs(Combination([apple,orange,pear]));
allstructs(Permutation([apple,orange,pear]), size=2);
allstructs(Permutation([apple,orange,pear]));
```

You can use **count** to count the number of objects of a given size

```
count(Combination([apple,orange,pear]),size=2);
count(Combination([apple,orange,pear]));
count(Permutation([apple,orange,pear]), size=2);
count(Permutation([apple,orange,pear]));
```

Remark. Notice the difference when using **Combination** vs. **Permutation**. Do you know why?