04 Counting Methods

04_02_Combinations

Combination. Let S be a set of $\{x_1, x_2, \ldots, x_n\}$. A r-combination based on S is a subset T of S which has a size r, where $0 \leq r \leq n$. The total number of r-combinations based S is denoted by C(n, r).

Example. Let S be {1, 2, 3, 4, 5}. Then the 0-combination based S is the empty set ∅. All the 1-combinations based on S are A_1 = {1}, A_2 = {2}, A_3 = {3}, A_4 ={4}, A_5 ={5}. All the 2-combinations based on S are B_1 ={1, 2}, B_2 = {1, 3}, B_3 ={1, 4}, B_4 = {1, 5}, B_5 = {2, 3}, B_6 = {2, 4}, B_7 = {2, 5}, B_8 = {3, 4}, B_9 = {3, 5}, B_{10} = {4, 5}. All the 3-combinations based on S are {3, 4, 5} = S - B_1, {2, 4, 5} = S - B_2, {2, 3, 5} = S - B_3, {2, 3, 4} = S - B_4, {1, 4, 5} = S - B_5, {1, 3, 5} = S - B_6, {1, 3, 4} = S - B_7, {1, 2, 5} = S - B_8, {1, 2, 4} = S - B_9, {1, 2, 3} = S - B_{10}. All the 4-combinations based on S are {2, 3, 4, 5} = S - A_1, {1, 3, 4, 5} = S - A_2, {1, 2, 4, 5} = S – A_3, {1, 2, 3, 5} = S - A_4, 1, 2, 3, 4} = S - A_5. Finally, the 5-combination based S is {1, 2, 3, 4, 5} = S - ∅. Thus C(5, 0) = C(5, 5) = 1, C(5, 1) = C(5, 4) = 5, and C(5, 2) = C(5, 3) = 10.

Theorem. $C(n, r) = P(n, r)/r! = n!/((n – r)!\, r!)$, where $0 \leq r \leq n$.

[Proof] Let T be a task of generating all the r-permutations based a set S of size n. T can be decomposed into a subtask T_1 of finding all the r-combinations based on S and a subtask T_2 of permutating the r elements in each r-combinations found in T_1. Obviously, there are P(n, r) ways for completing T, C(n, r) ways for completing T_1, and r! ways for completing T_2. By the Product Rule, we have

$$P(n, r) = C(n,r)*r!.$$

Thus

$$C(n, r) = P(n, r)/r! = n!/((n – r)!\ r!).$$

Corollary. $C(n, r) = C(n, n – r) = n!/((n – r)!\ r!)$, where $0 \le r \le n$.

[Proof1] $C(n, n – r) = n!/((n – (n – r))!\ (n – r)!) = n!/((n – r)!\ r!)$ $C(n, r)$.

[Proof2] Let A be the set of all the r-combinations based a set S of size n. Notice that each element in A is a subset of S with size r. Let B be the set of all the (n – r)-combinations based the same set S above. Notice that each element in B is a subset of S with size (n – r). Taking the complement of each element in A, we get a unique element in B. Taking the complement of each element in B, we get a unique element in A. Therefore, there exists a one-to-one correspondence between set A and set B. Hence, |A| = |B|. Clearly, |A| = C(n, r) and |B| = C(n, n – r). So

$$C(n, r) = C(n, n- r), \text{ where } 0 \le r \le n.$$

Notice that $C(n, 0) = C(n, n) = 1$ if $n \ge 0$. $C(n, 1) = C(n, n - 1) = n$ if $n \ge 1$.

Example. Suppose there are 17 faculty members in a computer science department. How many ways are there to form a recruiting committee consisting of 5 faculty members.

[Solution] C(17, 5) = 17!/((17 – 5)! 5!) = 6,188.

Example. Suppose there are 10 mathematical faculty members and 8 computer science faculty members in a Department of Mathematics and Computer Science. How many ways are there to form a committee on developing a Data Science program such that the committee consisting of 3 mathematical faculty members and 4 computer science faculty members.

[Solution] C(10, 3)*C(8, 4) = 10!/((10 – 3)! 3!)* 8!/((8 – 4)! 4!)

= 120*70 = 8400.

Example. Find the number of bit strings of length 7 having exactly 3 ones.

[Solution] C(7, 3) = 7!/((7 – 3)! 3!) = 35.

In general, the number of bit strings of length n having exactly r ones is C(n, r).

Suppose S is a set with size n. Then C(n, r) is the number of all subsets of S with size r = the number of bit strings of length n having exactly r ones. Then

The number of all the subsets of S =

the number of all subsets of S with size 0 +

the number of all subsets of S with size 1 +

the number of all subsets of S with size 2 +

…………

the number of all subsets of S with size (n – 1) +

the number of all subsets of S with size n =

the number of bit strings of length n having exactly 0 ones +

the number of bit strings of length n having exactly 1 one +

the number of bit strings of length n having exactly 2 ones +

…………

the number of bit strings of length n having exactly (n – 1) ones +

the number of bit strings of length n having exactly n ones =

the number of bit strings of length n = 2^n. Thus, we have

$$C(n, 0) + C(n, 1) + \dots + C(n, n) = 2^n$$

Note that the number of all the subsets of S is $|P(S)| = |2^S|$, where $P(S) = 2^S$ is the power set of S. Thus $|P(S)| = |2^S| = 2^{|S|} = 2^n$ if S is a finite set with size n.