Permutations vs. Combinations

- Both are ways to count the possibilities
- The difference between them is whether order matters or not
- Consider a poker hand:
  - A♣, 5♥, 7♠, 10♦, K♠
  - Is that the same hand as:
    - K♠, 10♦, 7♠, 5♥, A♣
- Does the order the cards are handed out matter?
  - If yes, then we are dealing with permutations
  - If no, then we are dealing with combinations

Combinations

- What if order doesn’t matter?
- In poker, the following two hands are equivalent:
  - A♠, 5♥, 7♠, 10♦, K♠
  - K♠, 10♦, 7♠, 5♥, A♣
- The number of r-combinations of a set with n elements, where n is non-negative and 0 ≤ r ≤ n is:
  \[ C(n, r) = \frac{n!}{r!(n-r)!} \]

Combinations example

- How many different poker hands are there (5 cards)?
  \[ C(52, 5) = \frac{52!}{5!(52-5)!} = \frac{52!}{5!47!} = \frac{52 \times 51 \times 50 \times 49 \times 48}{5 \times 4 \times 3 \times 2 \times 1} = 2,598,960 \]
- How many different (initial) blackjack hands are there?
  \[ C(52, 2) = \frac{52!}{2!(52-2)!} = \frac{52!}{2!50!} = \frac{52 \times 51}{2 \times 1} = 1,326 \]

Combination formula proof

- Let C(52, 5) be the number of ways to generate unordered poker hands
- The number of ordered poker hands is \( P(52, 5) = \frac{52!}{(52-5)!} = 2,598,960 \) 
- The number of ways to order a single poker hand is \( P(5,5) = 5! = 120 \)
- The total number of unordered poker hands is the total number of ordered hands divided by the number of ways to order each hand
- Thus, \( C(52, 5) = P(52,5)/P(5,5) \)
Combination formula proof

- Let \( C(n, r) \) be the number of ways to generate unordered combinations.
- The number of ordered combinations (i.e., \( r \)-permutations) is \( P(n, r) \).
- The number of ways to order a single one of those \( r \)-permutations is \( P(r, r) \).
- The total number of unordered combinations is the total number of ordered combinations (i.e., \( r \)-permutations) divided by the number of ways to order each combination.
- Thus, \( C(n, r) = P(n, r)/P(r, r) \).

Combination Formula

\[
C(n, r) = \frac{P(n, r)}{P(r, r)} = \frac{n!/(n-r)!}{r!/(r-r)!} = \frac{n!}{r!(n-r)!}
\]

Bit Strings

- How many bit strings of length 10 contain:
  - Exactly four 1’s?
    - Find the positions of the four 1’s
    - Does the order of these positions matter?
    - Thus, the answer is \( C(10, 4) = 210 \)
  - At most four 1’s?
    - There can be 0, 1, 2, 3, or 4 occurrences of 1
    - \( C(10, 0) + C(10, 1) + C(10, 2) + C(10, 3) + C(10, 4) \)
    - \( = 1 + 10 + 45 + 120 + 210 \)
    - \( = 386 \)

Corollary 1

- Let \( n \) and \( r \) be non-negative integers with \( r \leq n \). Then \( C(n, r) = C(n, n-r) \)
- Proof:
  \[
  C(n, r) = \frac{n!}{r!(n-r)!} = \frac{n!}{(n-r)!(n-(n-r))!} = \frac{n!}{r!(n-r)!}
  \]

Corollary example

- There are \( C(52, 5) \) ways to pick a 5-card poker hand.
- There are \( C(52, 47) \) ways to pick a 47-card hand.
- \( P(52, 5) = 2,598,960 = P(52, 47) \).
- When dealing 47 cards, you are picking 5 cards to not deal
  - As opposed to picking 5 card to deal
  - Again, the order the cards are dealt in does matter.
Note

• An alternative (and more common) way to denote an \(r\)-combination:

\[
\binom{n}{r} = \frac{n!}{r!(n-r)!}
\]

Choosing Teams

• Choosing team of 5 among 12
  
  • Two members must work as a pair
    
    - # of teams that contain both: \(C(10, 3) = 120\)
    
    - # of teams that don’t: \(C(10, 5) = 252\)
    
    - addition rule
  
  • Two members must be kept apart
    
    - # of teams that have either: \(2!C(10, 4) = 420\)
    
    - # of teams that don’t: \(C(10, 5) = 252\)

Choosing Teams

• We have 5 men and 7 women
  
  • How many 5-person groups can be chosen that
    
    - consist of 3 men and 2 women?
      
      \(C(5, 3) \times C(7, 2) = 210\)
    
    - have at least one man?
      
      \(C(12, 5) - C(7, 5) = 771\)
    
    - at most one man?
      
      \(C(7, 5) + C(5, 1) \times C(7, 4) = 196\)

\(r\)-Combinations with Repetitions

• How many 2-combinations can be selected from \(\{1, 2, 3\}\), if repetitions are allowed?
  
  \(\{1, 1\}, \{1, 2\}, \{1, 3\}, \{2, 2\}, \{2, 3\}, \{3, 3\}\)

<table>
<thead>
<tr>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
<th>selection</th>
<th>string</th>
</tr>
</thead>
<tbody>
<tr>
<td>XX</td>
<td></td>
<td></td>
<td>(1,1)</td>
<td>xx</td>
</tr>
<tr>
<td>X X</td>
<td></td>
<td></td>
<td>(1,2)</td>
<td>x</td>
</tr>
<tr>
<td>X X</td>
<td></td>
<td></td>
<td>(1,3)</td>
<td>x</td>
</tr>
<tr>
<td>XX</td>
<td></td>
<td></td>
<td>(2,2)</td>
<td>xx</td>
</tr>
<tr>
<td>X X</td>
<td></td>
<td></td>
<td>(2,3)</td>
<td>x</td>
</tr>
<tr>
<td>XX</td>
<td></td>
<td></td>
<td>(3,3)</td>
<td>xx</td>
</tr>
</tbody>
</table>

\(r\)-Combinations with Repetitions

• Strings of 4 symbols with 2 x’s and 2 |’s
  
  • Notice that once the positions of the x’s are fixed, the |’s just go between
  
  \(C(4, 2) = 4 \times 3 / 2 = 6\)

\(r\)-Combinations with Repetitions

• The number of \(r\)-Combinations with repetition allowed that can be selected from a set of \(n\) elements is: \(C(r+n-1, r)\)
Soda Distribution

- Select 15 cans of soft drinks from 5 types
  - How many different selections?
    - \( C(5+15-1, 15) = C(19, 15) = 3,876 \)
  - If Diet Coke is one of the types, how many selections include at least 6 cans Diet Coke?
    - choose the DCs first, then the rest
      - \( C(5+9-1, 9) = C(13, 9) = 715 \)
  - If the store only has 5 cans of DC, but at least 15 cans of all others, how many selections?

Circular seatings

- How many ways are there to seat 6 people around a circular table, where seatings are considered to be the same if they can be obtained from each other by rotating the table?
  - First, place the first person in the north-most chair
    - Only one possibility
  - Then place the other 5 people
    - There are \( P(5, 5) = 5! = 120 \) ways to do that
  - By the product rule, we get \( 1 \cdot 120 = 120 \)
  - Alternative means to answer this:
    - There are \( P(6, 6) = 720 \) ways to seat the 6 people around the table
    - For each seating, there are 6 "rotations" of the seating
    - Thus, the final answer is \( 720/6 = 120 \)

Ways to Count

- Choosing \( k \) elements from \( n \)

<table>
<thead>
<tr>
<th>Repetition allowed</th>
<th>Order matters</th>
<th>Order doesn't matter</th>
</tr>
</thead>
<tbody>
<tr>
<td>( n^k )</td>
<td>( n )</td>
<td>( C(k+n-1, k) )</td>
</tr>
<tr>
<td>No repetition</td>
<td>( P(n, k) )</td>
<td>( C(n, k) )</td>
</tr>
</tbody>
</table>

Horse races

- How many ways are there for 4 horses to finish if ties are allowed?
  - Note that order does matter!
  - Solution by cases
    - No ties
      - The number of permutations is \( P(4, 4) \times 4! = 24 \)
    - Two horses tie
      - There are \( C(4-2, 6 \) ways to choose the two horses that tie
      - There are \( P(4, 2) \) ways for the "groups" to finish
      - A "group" is either a single horse or the two tying horses
      - By the product rule, there are \( 6 \times 24 = 144 \) possibilities for this case
    - Two groups of two horses tie
      - There are \( C(4-2, 2 \times 2) \) ways to choose the two winning horses
      - The other two horses tie for second place
      - Three horses tie with each other
        - There are \( C(4-2, 3 \) ways to choose the three horses that tie
        - There are \( P(2, 2) \) ways for the "groups" to finish
      - By the product rule, there are \( 24 \times 2 \) possibilities for this case
    - All four horses tie
      - There is only one combination for this
  - By the sum rule, the total is \( 24 \times 36 + 8 = 1 = 75 \)

Counting Triples

- How many \( (i, j, k) \) such that \( 1 \leq i \leq j \leq k \leq n \)?
  - If \( n=5 \), represent \( (3, 3, 4) \) as \( ||x|x|\)
  - If \( n=7 \), represent \( (2, 4, 5) \) as \( |x||x|x|\)
  - How many \( |x| \)?
    - How many \( x| \)?
    - \( C(3+n-1, 3) = (n+2)!/3!x(n-1)! = (n+2)(n+1)n/6 \)

Nested for loop

- How many times will the innermost loop body be executed?
  - For each iteration, there is a different combination of the indices \( (i, j, k) \), \( 1 \leq i \leq j \leq k \leq n \)

```java
for (k := 1 to n)
  for (j := 1 to k)
    for (i := 1 to j)
      //body
      next i
      next j
      next k
```