Kombinatorik

Tim Olimpiade Komputer Indonesia

Pendahuluan

Melalui dokumen ini, kalian akan:

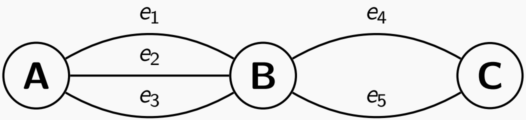
- Mempelajari **aturan perkalian** dan **aturan penjumlahan**.
- Mempelajari **permutasi**.
- Mempelajari **kombinasi**.
- Memahami **segitiga pascal**.

Bagian 1

Aturan Perkalian dan Aturan Penjumlahan

Permasalahan 1

- Terdapat 3 buah kota yaitu A, B, dan C.
- Kota A dan kota B terhubung oleh 3 jalur berbeda yaitu e_1, e_2, dan e_3.
- Sedangkan kota B dan kota C terhubung oleh 2 jalur berbeda yaitu e_4 dan e_5.
- Berapa banyak cara berbeda untuk menuju kota C dari kota A?
- Ilustrasi:



Solusi Awal

- Apabila kita hitung satu per satu, maka cara yang berbeda untuk menuju kota C dari kota A adalah sebagai berikut:
  - Melalui *jalur* e_1 kemudian *jalur* e_4
  - Melalui *jalur* e_1 kemudian *jalur* e_5
  - Melalui *jalur* e_2 kemudian *jalur* e_4
  - Melalui *jalur* e_2 kemudian *jalur* e_5
  - Melalui *jalur* e_3 kemudian *jalur* e_4
  - Melalui *jalur* e_3 kemudian *jalur* e_5
- Dengan kata lain, terdapat 6 cara berbeda untuk menuju kota C dari kota A.
- Tetapi, apabila jumlah kota dan jalur yang ada sangatlah banyak, kita tidak mungkin menulis satu per satu cara yang berbeda. Karena itulah kita gunakan **Aturan Perkalian**.

Aturan Perkalian

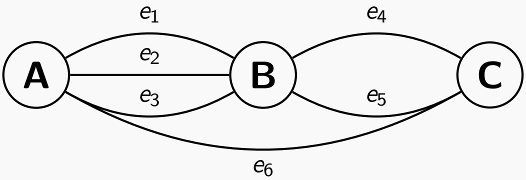
- Apabila suatu proses dapat dibagi menjadi N *sub-proses berbeda* yang mana terdapat a_i cara untuk menyelesaikan sub-proses ke-i.
- Banyak cara yang berbeda untuk menyelesaikan proses tersebut adalah $a_1 \times a_2 \times a_3 \times ... \times a_i$.

Solusi dengan Aturan Perkalian

- Jika kita anggap bahwa banyak cara dari kota A menuju kota B merupakan sub-proses pertama, yang mana terdapat 3 cara untuk menyelesaikan sub-proses tersebut.
- Kita anggap pula bahwa banyak cara dari kota B menuju kota C merupakan sub-proses kedua, yang mana terdapat 2 cara untuk menyelesaikan sub-proses tersebut.
- Karena perjalanan dari kota A menuju kota B dan dari kota B menuju kota C merupakan 2 sub-proses yang berbeda, maka kita dapat gunakan aturan perkalian.
- Banyak cara berbeda dari kota A menuju kota C adalah $3 \times 2 = 6$.

Permasalahan 2

- Permasalahan ini merupakan lanjutan dari **Permasalahan 1**. Deskripsi permasalahan, jumlah kota dan jalur serta susunan jalur yang ada sama persis dengan permasalahan tersebut.
- Apabila kita tambahkan 1 jalur lagi, yaitu e_6 yang menghubungkan kota A dan C.
- Berapa banyak cara berbeda untuk menuju kota C dari kota A?
- Ilustrasi:



Analisa Permasalahan

- Dengan menulis satu per satu setiap cara, maka terdapat 7 cara yang berbeda, yaitu 6 cara sesuai dengan permasalahan sebelumnya, ditambah dengan *menggunakan jalur* e_6.
- Apabila kita menggunakan Aturan Perkalian, maka didapatkan banyak cara yang berbeda adalah $3 \times 2 \times 1 = 6$ yang mana solusi tersebut tidaklah tepat.
- Kita tidak dapat menggunakan Aturan Perkalian dalam permasalahan ini, dikarenakan antara perjalanan dari kota A menuju kota C melalui kota B dengan tanpa melalui kota B merupakan 2 proses yang berbeda.
- Oleh karena itu, kita dapat menggunakan **Aturan Penjumlahan**.

Aturan Penjumlahan

- Apabila suatu proses dapat dibagi menjadi N *himpunan proses berbeda* yaitu $H_1, H_2, H_3, ..., H_N$ dengan setiap himpunannya *saling lepas* (tidak beririsan).
- Banyak cara yang berbeda untuk menyelesaikan proses tersebut adalah $|H_1| + |H_2| + |H_3| + ... + |H_i|$ dengan $|H_i|$ merupakan banyaknya cara berbeda untuk menyelesaikan proses ke-i.

Solusi dengan Aturan Penjumlahan

- Proses perjalanan dari kota A menuju kota C dapat kita bagi menjadi 2 himpunan proses yang berbeda, yaitu dari kota A menuju kota C melalui kota B, dan dari kota A langsung menuju kota C.
- Banyak cara dari kota A menuju kota C melalui kota B dapat kita dapatkan dengan Aturan Perkalian seperti yang dibahas pada permasalahan sebelumnya, yaitu 6 cara berbeda.
- Banyak cara dari kota A langsung menuju kota C adalah 1 cara, yaitu melalui jalur e_6.
- Dengan *Aturan Penjumlahan*, banyak cara berbeda dari kota A menuju kota C adalah $6 + 1 = 7$ cara berbeda.

Hati-Hati!

- Apabila terdapat irisan dari himpunan proses tersebut, maka solusi yang kita dapatkan dengan Aturan Penjumlahan menjadi tidak tepat, dikarenakan solusi dalam irisan tersebut terhitung 2 kali.
- Agar solusi tersebut menjadi tepat, gunakan Prinsip Inklusi-Eksklusi pada Teori Himpunan.

Bagian 2

Permutasi

Notasi Faktorial

- Perlu diperhatikan bahwa faktorial dari N ($N!$) merupakan hasil perkalian dari semua bilangan asli kurang dari atau sama dengan N.
- $N! = N \times (N - 1) \times (N - 2) \times ... \times 3 \times 2 \times 1$, dengan $O! = 1$.
- Contoh: $5! = 5 \times 4 \times 3 \times 2 \times 1 = 120$.

Redundansi

- Redundansi sering disebut juga dengan *Aturan Pembagian*.
- Apabila terdapat K susunan cara yang kita anggap berbeda dan seharusnya merupakan 1 cara yang sama, maka kita dapat membagi total keseluruhan cara dengan K, sehingga K cara tersebut dianggap sama sebagai 1 cara.

Contoh Redundansi

- Banyak kata berbeda yang disusun dari huruf-huruf penyusun "TOKI" adalah 4! (menggunakan aturan perkalian).
- Apabila kita ganti soal tersebut, yaitu kata berbeda yang disusun dari huruf-huruf penyusun "BACA", solusi 4! merupakan solusi yang salah.
- Hal ini disebabkan karena terdapat 2 buah huruf 'A'. Sebagai contoh: BA_1CA_2 dan BA_2CA_1 pada dasarnya merupakan kata yang sama.

Contoh Redundansi (lanj.)

- Terdapat 2! cara yang kita anggap berbeda tetapi seharusnya merupakan cara yang sama, yaitu penggunaan $A_1 A_2$ dan $A_2 A_1$.
- Sehingga banyak kata berbeda yang dapat kita bentuk dari huruf-huruf penyusun kata "BACA" adalah $\frac{4!}{2!} = \frac{24}{2} = 12$ kata berbeda.

Permasalahan 1

- Terdapat 5 anak (sebut saja A, B, C, D, dan E) yang sedang mengikuti sebuah kompetisi.
- Dalam kompetisi tersebut akan diambil 3 peserta sebagai pemenang.
- Berapa banyak susunan pemenang yang berbeda dari kelima orang tersebut?

Solusi Awal

- Anggap bahwa kita mengambil semua anak sebagai pemenang, sehingga terdapat $5! = 120$ susunan pemenang yang berbeda (ABCDE, ABCED, ABDCE, ..., EDCBA).
- Apabila kita hanya mengambil 3 peserta saja, perhatikan bahwa terdapat 2 cara yang kita anggap berbeda yang mana seharusnya merupakan cara yang sama. Contoh: (ABC)DE dan (ABC)ED merupakan cara yang sama, karena 3 peserta yang menang adalah A, B, dan C.
- Dengan menggunakan prinsip *Redundansi*, maka banyak susunan pemenang yang berbeda adalah $\frac{5!}{2!} = \frac{120}{2} = 60$ susunan berbeda.

Solusi Secara Umum

- Apabila terdapat N anak, dan kita mengambil semua anak sebagai pemenang, maka terdapat $N!$ susunan cara berbeda.
- Tetapi apabila kita hanya mengambil R anak saja, maka akan terdapat $(N-R)!$ susunan yang kita anggap berbeda yang mana seharusnya merupakan satu cara yang sama.
- Secara umum dengan prinsip Redundansi, banyak susunan berbeda adalah $\frac{N!}{(N-R)!}$.
- Rumus itulah yang kita kenal dengan istilah **Permutasi**.

Permutasi

- Apabila terdapat *n objek* dan kita akan mengambil *r objek* dari n objek tersebut yang mana $r < n$ dan *urutan pengambilan diperhitungkan*.
- Banyak cara pengambilan yang berbeda adalah Permutasi r terhadap n:
- $P(n,r) =_n P_r = P^n_r = \frac{n!}{(n-r)!}$.

Permasalahan 2

- Permasalahan ini sejenis dengan contoh pada Redundansi.
- Berapa banyak kata berbeda yang disusun dari huruf-huruf penyusun kata "MEGAGIGA"?

Solusi Awal

- Terdapat 8 huruf, sehingga banyak kata yang dapat disusun adalah 8!.
- Terdapat 3 huruf 'G' sehingga terdapat 6 kata yang kita anggap berbeda $(G_1G_2G_3, G_1G_3G_2, ..., G_3G_2G_1)$ yang mana seharusnya keenam kata tersebut merupakan kata yang sama.
- Dengan prinsip Redundansi, maka banyak kata yang dapat disusun mengingat kesamaan kata pada huruf G adalah $\frac{8!}{3!}$.
- Perlu kita perhatikan pula bahwa terdapat 2 huruf A, sehingga dengan cara yang sama akan didapatkan banyak kata yang berbeda adalah $\frac{8!}{(3! \times 2!)}$.

Solusi Secara Umum

- Terdapat N huruf, sehingga banyak kata yang dapat kita susun adalah $N!$.
- Apabila terdapat K huruf dengan setiap hurufnya memiliki R_i huruf yang sama, maka dengan prinsip Redundansi banyak kata berbeda yang dapat disusun adalah $\frac{N!}{(R_1! \times R_2! \times R_3! \times \ldots \times R_K!)}$.
- Rumus itulah yang kita kenal dengan **Permutasi Elemen Berulang**.

Permutasi Elemen Berulang

- Apabila terdapat n objek dan terdapat k objek yang mana setiap objeknya memiliki r_i elemen yang berulang, maka banyaknya cara berbeda dalam menyusun objek tersebut adalah:
- $P^n_{r_1,r_2,r_3,...,rk} = \frac{n!}{(r_1! \times r_2! \times r_3! \times ... \times r_k!)}$
- Syarat untuk menggunakan rumus tersebut adalah jika $r_1 + r_2 + r_3 + ... + r_k \leq n$.

Permasalahan 3

- Terdapat 4 anak, sebut saja A, B, C, dan D.
- Berapa banyak susunan posisi duduk yang berbeda apabila mereka duduk melingkar?

Solusi Awal

- Banyak susunan posisi duduk yang berbeda apabila mereka duduk seperti biasa (tidak melingkar) adalah $4! = 120$.
- Perhatikan bahwa posisi duduk ABCD, BCDA, CDAB, dan DABC merupakan susunan yang sama apabila mereka duduk melingkar, dikarenakan susunan tersebut merupakan rotasi dari susunan yang lainnya.
- Dengan kata lain terdapat 4 cara yang kita anggap berbeda yang mana seharusnya hanya dihitung sebagai 1 cara yang sama.
- Dengan prinsip Redundansi, banyak susunan posisi duduk yang berbeda adalah $\frac{120}{4} = 30$ susunan berbeda.

Solusi Secara Umum

- Banyak susunan posisi duduk yang berbeda apabila N anak seperti biasa (tidak melingkar) adalah $N!$.
- Dengan analisa yang sama, akan terdapat N cara yang kita anggap berbeda (pada posisi duduk tidak melingkar) yang mana seharusnya dihitung sebagai 1 cara yang sama (pada posisi duduk melingkar).
- Dengan prinsip Redundansi, banyak susunan posisi duduk yang berbeda adalah $\frac{N!}{N} = (N-1)!$ susunan berbeda.
- Rumus itulah yang kita kenal dengan istilah **Permutasi Siklis**.

Permutasi Siklis

- Permutasi Siklis adalah Permutasi yang disusun melingkar.
- Banyaknya susunan yang berbeda dari Permutasi siklis terhadap n objek adalah:
- $P^{n}_{(siklis)} = (n-1)!$

Bagian 3

Kombinasi

Permasalahan 1

- Terdapat 5 siswa (sebut saja A, B, C, D, dan E) yang mana akan dipilih 3 siswa untuk mengikuti kompetisi.
- Berapa banyak susunan tim berbeda yang dapat dibentuk?

Solusi Awal

- Solusi ini berbeda dengan Permutasi, karena susunan ABC dan BAC merupakan susunan yang sama, yaitu 1 tim terdiri dari A, B, dan C.
- Apabila kita anggap bahwa mereka merupakan susunan yang berbeda, maka banyak susunan tim adalah $P_2^5 = \frac{120}{2} = 60$.
- Untuk setiap susunan yang terdiri dari anggota yang sama akan terhitung 6 susunan berbeda yang mana seharusnya hanya dihitung sebagai 1 susunan yang sama.
- Contoh: ABC, ACB, BAC, BCA, CAB, CBA merupakan 1 susunan yang sama.
- Oleh karena itu dengan prinsip Redundansi, banyaknya susunan tim yang berbeda adalah $\frac{60}{6} = 10$ susunan berbeda.

Solusi Secara Umum

- Apabila terdapat N siswa dan akan kita ambil R siswa untuk dibentuk sebagai 1 tim, maka banyak susunan berbeda adalah sebagai berikut:
- Apabila kita anggap bahwa mereka merupakan susunan yang berbeda, maka banyak susunan tim adalah P_R^N.
- Untuk setiap susunan yang terdiri dari anggota yang sama akan terhitung R! susunan berbeda yang mana seharusnya hanya dihitung sebagai 1 susunan yang sama.
- Oleh karena itu dengan prinsip Redundansi, banyaknya susunan tim yang berbeda adalah $\frac{P_R^N}{R!} = \frac{N!}{(N-R)! \times R!}$ susunan berbeda.
- Rumus itulah yang kita kenal dengan istilah **Kombinasi**.

Kombinasi

- Apabila terdapat *n objek* dan kita akan mengambil *r objek* dari n objek tersebut dengan $r < n$ dan *urutan pengambilan tidak diperhitungkan*.
- Banyak cara pengambilan yang berbeda adalah Kombinasi r terhadap n:
- $C(n, r) =_n C_r = C_r^n = \frac{n!}{(n-r)! \times r!}$.

Permasalahan 2

- Pak Dengklek ingin membeli kue pada toko kue, yang mana toko tersebut menjual 3 jenis kue, yaitu rasa coklat, strawberry, dan kopi.
- Apabila Pak Dengklek ingin membeli 4 buah kue, maka berapa banyak susunan kue berbeda yang Pak Dengklek dapat beli?

Analisa Permasalahan

- Dapat kita lihat bahwa jika kita membeli coklat-strawberry dengan strawberry-coklat akan menghasilkan susunan yang sama, dengan kata lain permasalahan ini merupakan permasalahan *Kombinasi*.
- Akan tetapi, kita dapat membeli suatu jenis kue beberapa kali atau bahkan tidak membeli sama sekali.
- Apabila kita sederhanakan permasalahan tersebut dalam model matematika, maka permasalahan tersebut menjadi:
- Banyaknya kemungkinan nilai A, B, dan C yang mana $A + B + C = 4$ dan $A, B, C \geq 0$.

Solusi

- Kita dapat membagi 4 kue tersebut menjadi 3 bagian. Untuk mempermudah ilustrasi tersebut, kita gunakan lambang o yang berarti kue, dan | yang berarti pembatas.
- Bagian kiri merupakan kue A, bagian tengah merupakan kue B, dan bagian kanan merupakan kue C.
- Contoh: $(o|o|oo)$ melambangkan 1 kue A, 1 kue B, dan 2 kue C.
- Contoh Lain: $(oo|oo|)$ melambangkan 2 kue A, 2 kue B, dan 0 kue C.
- Dengan kata lain, semua susunan yang mungkin adalah $(oooo||)$, $(ooo|o|)$, $(oo|oo|)$, ..., $(||oooo)$ yang tidak lain merupakan $C^6_2 = \frac{6!}{4! \times 2!} = 15$ susunan berbeda.

Solusi secara umum

- Berapa banyak susunan nilai berbeda dari $X_1, X_2, X_3, ..., X_n$ yang mana $X_1 + X_2 + X_3 + ... + X_n = r$.
- Untuk membagi n kue tersebut menjadi r bagian, maka akan dibutuhkan r-1 buah pembatas, sehingga akan terdapat n+r-1 buah objek, yang mana kita akan memilih n-1 objek untuk menjadi simbol 'o'.
- Dengan kata lain, banyaknya susunan nilai yang berbeda adalah C_{n-1}^{n+r-1}.
- Rumus itulah yang kita kenal dengan istilah **Kombinasi dengan Pengulangan**.

Kombinasi dengan Pengulangan

- Terdapat n jenis objek yang mana kita akan mengambil r objek, dan *tiap objeknya dapat tidak diambil/diambil lebih dari 1 kali*.
- Banyak cara berbeda yang memenuhi syarat tersebut adalah sebagai berikut:
- $C^{n+r-1}_{n-1} = C^{n+r-1}_{r} = \frac{(n+r-1)!}{(n-1)! \times r!}$.
- Keterangan: dikarenakan setiap objek dapat diambil lebih dari 1 kali, maka r tidak harus kurang dari atau sama dengan n.

Bagian 4

Segitiga Pascal

Segitiga Pascal

- Segitiga Pascal merupakan susunan dari Koefisien Binomial dalam bentuk segitiga.
- Nilai dari baris ke-n suku ke-r adalah:
- $A_{ij} = C^n_r$.
- Contoh Segitiga Pascal:
  - Baris ke-1: 1
  - Baris ke-2: 1 1
  - Baris ke-3: 1 2 1
  - Baris ke-4: 1 3 3 1
  - Baris ke-5: 1 4 6 4 1

Analisa

- Diberikan suatu himpunan semesta $S = X_1, X_2, ..., X_n$. Berapa banyak cara berbeda apabila kita mengambil r objek dari S?
- Kita bagi 2 kasus, yaitu:
  - Kasus 1: Apabila kita menggunakan X_n, sehingga kita memilih r-1 objek dari himpunan $A = X_1, X_2, X_3, ..., X_{n-1}$, 1 objek sisanya merupakan X_n. Banyak cara berbeda dari kasus ini adalah C^{n-1}_{r-1}.
  - Kasus 2: Apabila kita tidak menggunakan X_n, sehingga kita memilih r objek dari himpunan $A = X_1, X_2, X_3, ..., X_n$, 1 objek sisanya merupakan X_n. Banyak cara berbeda dari kasus ini adalah C^{n-1}_{r}.

Analisa (lanj.)

- Pada dasarnya apabila kita menggunakan rumus kombinasi, maka kita akan memilih r objek dari n objek, sehingga banyak cara berbeda secara keseluruhan adalah C_r^n.
- Dengan aturan penjumlahan dari kasus 1 dan kasus 2, kita dapatkan $C_r^n = C_{r-1}^{n-1} + C_r^{n-1}$.
- Dengan kata lain, dalam koefisien binomial pada segitiga pascal, akan berlaku:
- $A_{(n)(r)} = A_{(n-1)(r-1)} + A_{(n-1)(r)}$.
- Persamaan itulah yang sering kita gunakan dalam membuat segitiga pascal.

Analisa (lanj.)

- Dalam dunia pemrograman, persamaan tersebut dapat kita gunakan untuk pre-komputasi semua nilai dari C_r^n yang nantinya akan kita butuhkan.
- Untuk mencari nilai dari C_r^n, karena kita perlu mencari nilai faktorial dari n, sehingga kompleksitasinya O(N).
- Apabila kita gunakan iterasi untuk mencari nilai dari C_r^n untuk pre-komputasi semua nilai kombinasi yang mungkin (yang mana kurang lebihnya akan ada N^2 perhitungan), kompleksitasnya akan menjadi $O(N^3)$.
- Dengan menggunakan persamaan $A_{(n)(r)} = A_{(n-1)(r-1)} + A_{(n-1)(r)}$ yang mana kompleksitasnya adalah O(1), maka secara keseluruhan kompleksitasnya akan menjadi $O(N^2)$ menjadikan solusi ini jauh lebih cepat.

Implementasi Segitiga Pascal

1. Kita gunakan iterasi untuk mencari semua nilai dari kombinasi r terhadap n.
2. Untuk menyimpan semua nilai dari kombinasi tersebut akan dibutuhkan array 2-dimensi yang mana A[n][r] menyimpan nilai kombinasi r terhadap n.
3. Nilai dari A[0][0] adalah 1.
4. Iterasi i dari 1 hingga n
    - 4.1 Iterasi j dari 0 hingga n
        - 4.1.1 Jika $j = 0$ atau $j = n$, maka $A[i][j] = 1$
        - 4.1.2 Selain itu, $A[i][j] = A[i-1][j-1] + A[i-1][j]$

Implementasi Segitiga Pascal (lanj.)

SEGITIGA_PASCAL(*baris*)

// Sediakan array 2-dimensi C berukuran $baris \times baris$
// Perhatikan bahwa index array C pada contoh ini dimulai dari 0.
$C[0][0] = 1$
for $i = 1$ **to** $baris$
    for $j = 0$ **to** $baris$
        if $j = 0$ *or* $j = baris$
            $C[i][j] = 1$
        else
            $C[n][r] = C[n-1][r-1] + C[n-1][r]$

Penggunaan Segitiga Pascal

- Dalam bidang matematika, segitiga pascal merupakan kumpulan dari koefisien binomial yang dapat digunakan dalam Binomial Newton $(x + y)^n = \sum_{r=0}^{n} a_r x^{n-r} y^r$ yang mana a_r merupakan bilangan dalam segitiga pascal baris ke-n suku ke-r.
- Dalam bidang programming, algoritma dari segitiga pascal dapat digunakan untuk mencari semua nilai dari kombinasi r terhadap n dengan kompleksitas waktu $O(N^2)$ dan memori yang dibutuhkan adalah $O(N^2)$.

Penutup

- Materi ini berisi mengenai Kombinatorik Dasar dan implementasinya yang umum digunakan dalam pemrograman kompetitif.
- Dengan materi ini, Anda diharapkan dapat menggunakan konsep kombinatorik yang ada.
- Selamat berlatih dan semoga sukses!