

Penerapan Program Dinamis untuk Optimalisasi Keuntungan Produksi Pesanan Baju Skala Besar

Makalah IF2211 Strategi Algoritma

Abdurrahman Adni (13517117)

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia

13517117@std.stei.itb.ac.id

Abstract— Program Dinamik adalah suatu teknik matematika yang digunakan untuk mengoptimalkan proses pengambilan keputusan secara bertahap-ganda. Dalam teknik ini, keputusan yang menyangkut suatu persoalan dioptimalkan secara bertahap dan bukan secara sekaligus. Jadi inti dari teknik ini adalah membagi satu persoalan atas beberapa bagian persoalan yang dalam program dinamik disebut tahap. Kemudian memecahkan tiap tahap dengan mengoptimalkan keputusan atas tiap tahap sampai seluruh persoalan telah terpecahkan. Keputusan yang optimal atas seluruh persoalan ialah kumpulan dari sejumlah keputusan optimal atas seluruh tahap yang kemudian disebut sebagai kebijakan optimal. Salah satu aplikasi atau penerapan dari program dinamis adalah untuk menghitung keuntungan yang akan didapat oleh sebuah pabrik konveksi pakaian yang menerima pesanan baju dengan skala besar.

Keywords—Program Dinamis; Keuntungan; Pabrik Konveksi; Optimal

I. PENDAHULUAN

Konveksi merupakan salah satu bisnis di bidang pakaian yang biasanya memproduksi pakaian dalam partai besar sesuai permintaan. Bisnis konveksi banyak diminati di Indonesia karena permintaan pasar yang tinggi namun kompetitornya masih rendah. Keberadaan pabrik konveksi sangat menunjang terhadap kemajuan industri pakaian jadi di Indonesia, karena selain mengerjakan pembuatan pakaian dari pemesan untuk pasar lokal, konveksi juga bisa menerima dan mengerjakan proses jahit sebuah pabrik garmen dalam pembuatan pakaian jadi skala besar untuk pasar lokal maupun pasar ekspor.

Masalah yang muncul kemudian adalah apakah pabrik konveksi ini mampu menerima dan mengerjakan segala permintaan konsumen dengan pesanan jumlah besar, dikerjakan dalam tenggat waktu yang terbatas, serta dapat menghasilkan keuntungan yang maksimal. Apabila pabrik terlambat dalam menyelesaikan permintaan, tentu akan menjadi kerugian yang besar bagi pabrik konveksi itu sendiri. Konsumer yang tidak puas karena keterlambatan tentu tidak akan melanjutkan kerjasama lagi dengan pabrik konveksi tersebut, serta nama baik pabrik juga akan rusak dan menurunkan jumlah peminat. Disisi lain, keuntungan juga harus diperhitungkan matang-

matang guna menghindari kerugian materil bagi pabrik. Untuk itu, diperlukan strategi yang baik agar pabrik konveksi dapat mencapai keuntungan yang optimal. Oleh karena itu, makalah ini dibuat untuk memecahkan permasalahan tersebut dengan menggunakan program dinamis.

Metode *Dynamic Programming* pertama kali dikembangkan oleh Richard E. Bellman pada tahun 1957. Teknik yang digunakan program dinamis adalah mengoptimalkan proses pengambilan keputusan secara bertahap-ganda. Dalam teknik ini, keputusan yang menyangkut suatu persoalan dioptimalkan secara bertahap dan bukan secara sekaligus. Jadi inti dari teknik ini adalah membagi satu persoalan atas beberapa bagian persoalan yang dalam program dinamik disebut tahap. Kemudian memecahkan tiap tahap dengan mengoptimalkan keputusan atas tiap tahap sampai seluruh persoalan telah terpecahkan. Keputusan yang optimal atas seluruh persoalan ialah kumpulan dari sejumlah keputusan optimal atas seluruh tahap yang kemudian disebut sebagai kebijakan optimal.

II. DASAR TEORI

A. Program Dinamis

Program Dinamis (*dynamic programming*) merupakan metode pemecahan masalah dengan cara menguraikan solusi menjadi sekumpulan tahapan (*stage*) sedemikian sehingga solusi dari persoalan dapat dipandang dari serangkaian keputusan yang saling berkaitan. Keputusan yang optimal atas seluruh persoalan ialah kumpulan dari sejumlah keputusan optimal atas seluruh tahap yang kemudian disebut sebagai kebijakan optimal.

Karakteristik penyelesaian persoalan dengan Program Dinamis:

1. Terdapat sejumlah berhingga pilihan yang mungkin,
2. Solusi pada setiap tahap dibangun dari hasil solusi tahap sebelumnya,
3. Gunakan persyaratan optimasi dan kendala untuk membatasi sejumlah pilihan yang harus dipertimbangkan pada suatu tahap.

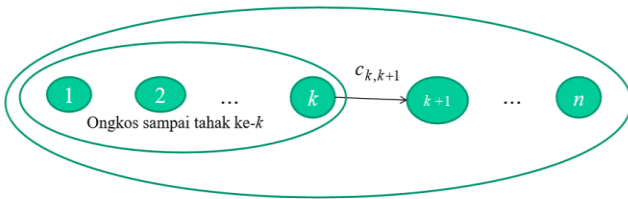
Terdapat perbedaan antara program dinamis dengan algoritma Greedy. Pada program Greedy, hanya satu rangkaian keputusan yang dihasilkan. Sedangkan pada program dinamis, terdapat lebih dari satu rangkaian keputusan yang dipertimbangkan.

A.1. Prinsip Optimalitas

- Pada program dinamis, rangkaian keputusan yang optimal dibuat dengan menggunakan Prinsip Optimalitas.

Prinsip Optimalitas
jika solusi total optimal, maka bagian solusi sampai tahap ke-k juga optimal.

- Prinsip optimalitas berarti bahwa jika kita bekerja dari tahap k ke tahap k + 1, kita dapat menggunakan hasil optimal dari tahap k tanpa harus kembali ke tahap awal.
- ongkos pada tahap k + 1 = (ongkos yang dihasilkan pada tahap k) + (ongkos dari tahap k ke tahap k + 1)

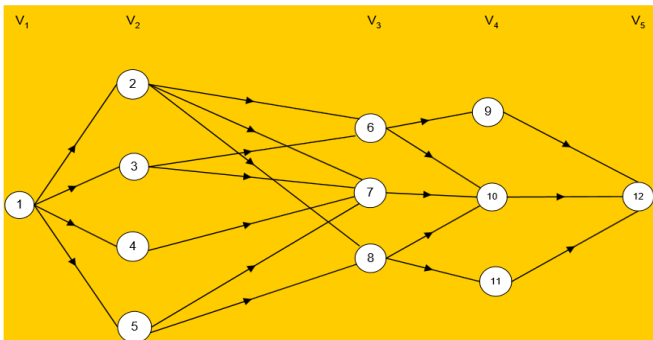


Gambar 1. Prinsip optimalitas

Sumber : Program-Dinamis-(2018), Rinaldi Munir, Slide 9

A.2. Karakteristik Persoalan Program Dinamis

1. Persoalan dapat dibagi menjadi beberapa tahap (stage), yang pada setiap tahap hanya diambil satu keputusan.
2. Masing-masing tahap terdiri dari sejumlah status (state) yang berhubungan dengan tahap tersebut. Secara umum, status merupakan bermacam kemungkinan masukan yang ada pada tahap tersebut. Graf multistage (*multistage graph*). Tiap simpul di dalam graf tersebut menyatakan status, sedangkan V_1, V_2, \dots menyatakan tahap.



Gambar 2. Prinsip optimalitas

Sumber : Program-Dinamis-(2018), Rinaldi Munir, Slide 11

3. Hasil dari keputusan yang diambil pada setiap tahap ditransformasikan dari status yang bersangkutan ke status berikutnya pada tahap berikutnya.
4. Ongkos (cost) pada suatu tahap meningkat secara teratur (steadily) dengan bertambahnya jumlah tahapan.
5. Ongkos pada suatu tahap bergantung pada ongkos tahap-tahap yang sudah berjalan dan ongkos pada tahap tersebut.
6. Keputusan terbaik pada suatu tahap bersifat independen terhadap keputusan yang dilakukan pada tahap sebelumnya.
7. Adanya hubungan rekursif yang mengidentifikasi keputusan terbaik untuk setiap status pada tahap k memberikan keputusan terbaik untuk setiap status pada tahap k + 1.
8. Prinsip optimalitas berlaku pada persoalan tersebut.

A.3. Dua pendekatan Program Dinamis

1. Program Dinamis Maju

Program dinamis maju. Program dinamis bergerak mulai dari tahap 1, terus maju ke tahap 2, 3, dan seterusnya sampai tahap n. Runtunan peubah keputusan adalah x_1, x_2, \dots, x_n .

Prinsip optimalitas pada PD maju :
ongkos pada tahap k + 1 = (ongkos yang dihasilkan pada tahap k) + (ongkos dari tahap k ke tahap k + 1) $k = 1, 2, \dots, n - 1$

2. Program Dinamis Mundur

Program dinamis ini bergerak mulai dari tahap n, terus mundur ke tahap n - 1, n - 2, dan seterusnya sampai tahap 1. Runtunan peubah keputusan adalah x_n, x_{n-1}, \dots, x_1 .

Prinsip optimalitas pada PD mundur:
ongkos pada tahap k = (ongkos yang dihasilkan pada tahap k + 1) + (ongkos dari tahap k + 1 ke tahap k) $k = n, n - 1, \dots, 1$

A.3. Langkah-langkah Pengembangan Algoritma Program Dinamis

Berikut adalah langkah-langkah pengembangan program dinamis :

1. Karakteristikkan struktur solusi optimal.
2. Definisikan secara rekursif nilai solusi optimal.
3. Hitung nilai solusi optimal secara maju atau mundur.
4. Konstruksi solusi optimal.

B. Teori Produksi

Teori produksi adalah studi tentang produksi atau proses ekonomi untuk mengubah faktor produksi (*input*) menjadi hasil produksi (*output*). Produksi menggunakan sumber daya untuk menciptakan barang atau jasa yang sesuai untuk digunakan. Dalam teori produksi, produksi adalah suatu kegiatan untuk menambah nilai guna pada suatu barang. Produksi di ukur sebagai “tingkat hasil produksi (*output*) perperiode waktu” karena merupakan konsep aliran.

Ada 3 aspek proses produksi antara lain :

- Kuantitas barang atau jasa di hasilkan.
- Bentuk barang atau jasa di ciptakan, dan
- Distribusi temporal dan spasial dari barang atau jasa yang di hasilkan.

B.1. Proses Produksi

Proses produksi dapat di definisikan sebagai kegiatan yang meningkatkan kesamaan antara pola permintaan barang atau jasa dan kuantitas, bentuk ukuran, panjang dan distribusi barang atau jasa tersedia bagi pasar. Dalam aktivitas produksinya produsen mengubah berbagai faktor produksi menjadi barang dan jasa. Berdasarkan hubungannya dengan tingkat produksi, faktor produksi dibedakan menjadi faktor produksi tetap dan faktor produksi variabel.

C. Teori Biaya

C.1. Konsep Biaya

Dalam konsep biaya, biaya dapat dibagi dua yaitu biaya eksplisit dan biaya implisit. Biaya eksplisit adalah biaya-biaya yang secara eksplisit terlihat, terutama melalui laporan keuangan. Biaya listrik, telepon dan air, demikian juga pembayaran demikian juga upah buruh dan gaji karyawan merupakan biaya eksplisit. Biaya implisit adalah biaya kesempatan.

Perilaku biaya juga berhubungan dengan periode produksi. Dalam jangka pendek ada faktor produksi tetap yang menimbulkan biaya tetap, yaitu biaya produksi yang besarnya tidak tergantung pada tingkat produksi. Dalam

jangka panjang, karena semua faktor variabel, biaya juga variabel (berubah-ubah)

Dalam jangka panjang, perusahaan akan lebih mudah meningkatkan produktivitas dibanding dalam jangka pendek. Itu sebabnya ada perusahaan yang mampu menekan biaya produksi, sehingga setiap tahun biaya produksi per unit makin rendah.

C.2. Biaya Produksi Jangka Pendek

Terdapat beberapa jenis biaya produksi jangka pendek dalam teori Biaya, diantaranya yaitu :

- Biaya Total, Biaya Tetap dan Biaya Variabel

Biaya total jangka pendek sama dengan biaya tetap ditambah biaya variabel. Biaya tetap adalah biaya yang besarnya tidak tergantung pada jumlah produksi, contohnya biaya barang modal, gaji pegawai, bunga pinjaman, sewa gedung kantor.

- Biaya Rata-rata

Biaya rata-rata adalah biaya yang harus dikeluarkan untuk memproduksi satu unit output. Besarnya biaya rata-rata adalah biaya total dibagi jumlah output.

- Biaya Marjial

Biaya marjial adalah tambahan biaya karena menambah sebanyak satu unit output.

D. Laba / Keuntungan

Menurut M. Nafarin (2007: 788) “Laba (*income*) adalah perbedaan antara pendapatan dengan keseimbangan biaya-biaya dan pengeluaran untuk periode tertentu”.

Menurut Abdul Halim & Bambang Supomo (2005:139) “Laba merupakan pusat pertanggungjawaban yang masukan dan keluarannya diukur dengan menghitung selisi antara pendapatan dan biaya”.

Menurut Mahmud M. Hanafi (2010:32), menyatakan bahwa “Laba merupakan ukuran keseluruhan prestasi perusahaan, yang didefinisikan sebagai berikut : Laba = Penjualan- Biaya”

Dapat disimpulkan, untung adalah kondisi dimana harga penjualan lebih besar dari pada harga pembelian.

Rumus Jumlah Keuntungan :

$$\text{Jumlah Keuntungan} = \text{Harga Penjualan} - \text{Biaya Produksi}$$

Penetapan margin keuntungan terhadap suatu produk yang akan dijual sangat dipengaruhi oleh biaya-biaya yang

dikeluarkan sampai barang tersebut diterima oleh konsumen. Produk yang sama bisa saja berbeda harganya di tempat yang berbeda, karena ada tambahan biaya pengiriman. Penetapan harga dengan menghitung biaya-biaya biasanya kita kenal dengan istilah *Cost Oriented Pricing*, dimana :

$$\text{HARGA JUAL} = \text{HARGA BELI} + \text{COST} + \text{MARK UP}$$

Selain itu, dikenal istilah *Demand Oriented Pricing*, yaitu suatu cara penetapan harga yang didasarkan pada banyaknya permintaan. Jika permintaan naik harga pun cenderung naik, dan sebaliknya jika permintaan turun maka harga cenderung turun walaupun mungkin biaya yang di keluarkan sama saja.

III. ANALISIS MASALAH DAN IMPLEMENTASI ALGORITMA

A. Studi Kasus Permasalahan

Studi kasus permasalahan ini akan mengambil contoh permintaan pembuatan baju skala besar dengan diberikan waktu mulai, estimasi tanggal selesai pengerjaan, dan keuntungan.

Sebagaimana yang telah dijelaskan sebelumnya, keuntungan didapat dengan menggunakan rumus $\text{KEUNTUNGAN} = \text{Harga Penjualan} - \text{Biaya Produksi}$. Serta harga penjualan didapat dengan rumus $\text{HARGA JUAL} = \text{HARGA BELI} + \text{COST} + \text{MARK UP}$. Serta ada juga yang dikenal dengan istilah *Demand Oriented Pricing*, yaitu suatu cara penetapan harga yang didasarkan pada banyaknya permintaan. Maka, diperlukan perhitungan khusus yang dalam hal ini akan dilakukan oleh ahli atau pekerja yang bekerja di bagian *marketing* dan keuangan. Oleh karena itu, *input* keuntungan per pesanan yang diterima oleh program adalah *input* keuntungan yang telah diolah sedemikian rupa sehingga *input* keuntungan merupakan keuntungan bersih per pesanan.

Input estimasi tanggal selesai pengerjaan yang akan diterima oleh program merupakan estimasi tanggal selesainya pekerjaan berdasarkan perkiraan bagian produksi untuk menyelesaikan satu paket pesanan baju.

Misal ada 5 permintaan pesanan baju sebagai berikut :

| | |
|-------------|----------------|
| Pesanan | : A |
| Start Time | : 2/4/2019 |
| End Time | : 5/4/2019 |
| Profit (Rp) | : 12.000.000,- |
| Pesanan | : B |
| Start Time | : 6/4/2019 |
| End Time | : 7/4/2019 |
| Profit (Rp) | : 8.000.000,- |
| Pesanan | : C |
| Start Time | : 7/4/2019 |
| End Time | : 9/4/2019 |
| Profit (Rp) | : 4.000.000,- |

| | |
|-------------|----------------|
| Pesanan | : D |
| Start Time | : 1/4/2019 |
| End Time | : 3/4/2019 |
| Profit (Rp) | : 10.000.000,- |
| Pesanan | : E |
| Start Time | : 5/4/2019 |
| End Time | : 8/4/2019 |
| Profit (Rp) | : 22.000.000,- |
| Pesanan | : F |
| Start Time | : 4/4/2019 |
| End Time | : 6/4/2019 |
| Profit (Rp) | : 10.000.000,- |

Berikut struktu data yang akan digunakan untuk menyelesaikan persoalan ini :

```

- struct Pesanan {
    int start_time;
    int end_time;
    long profit;
    char* name;
};
Pesanan adalah sebuah tipe data bentukan yang akan menyimpan record dari waktu mulai pesanan, tanggal selesai pesanan, dan keuntungan pesanan.
- int n          : jumlah pesanan
- Pesanan p[n]  : sebagai array of Pesanan sejumlah
- Int max_prof  : profit maksimal
    
```

B. Analisis dan Penyelesaian

Untuk menyelesaikan masalah optimasi ini, maka akan digunakan algoritma program dinamis dengan pendekatan heuristik untuk memaksimalkan keuntungan / *profit* tanpa menyebabkan *overlapping* pekerjaan. Berikut adalah langkah-langkah pendekatan program dinamis untuk permasalahan ini :

- 1) Urutkan pesanan berdasarkan *end_time* dari yang terkecil hingga terbesar.
- 2) Tambahkan sebuah *temporary array temp_prof* sebesar n (n adalah jumlah pesanan). **temp_prof** akan menampung akumulasi profit maksimal dari setiap pesanan yang dikerjakan.

- 3) Dengan melakukan pengulangan dari $i = 0$ sampai dengan $n-1$, inialisasi **temp_prof[i] = p[i].profit**.
- 4) Profit pesanan layak untuk ditambah satu sama lain yaitu profit dari pesanan yang tidak saling *overlapping*. Berikut implementasinya :
Untuk *integer* $i = 1$ sampai n :
Untuk *integer* $j = 0$ sampai $i-1$:
Jika $p[j].end_time \leq p[i].end_time$:
Jika $temp_prof[j] + p[i].profit > temp_prof[i]$:
 $temp_prof[i] = temp_prof[j] + p[i].profit$
- 5) Selanjutnya, untuk mendapat cari nilai maksimum dari *array* temp_prof, buat sebuah *variable* max_profit yang bertipe integer. Inialisasi max_profit = 0. lakukan perulangan $i = 0$ sampai n . Jika $temp_prof[i] > max_profit$, maka $max_profit = temp_prof[i]$
- 6) Itulah nilai profit maksimum yang merupakan keuntungan terbaik dari permasalahan ini.
- 7) Untuk melihat pesanan-pesanan mana saja yang menghasilkan keuntungan terbaik, bisa digunakan strategi berikut.
 - Buat stack terlebih dahulu, misal : S
 - Inialisasi $i = n-1$
 - Ketika $i \geq 0$ dan $max_profit > 0$:
S.push(p[i].name)
 $max_profit = max_profit - p[i].profit$
 $i = i-1$
 - Terakhir, secara iteratif *print/cetak* S.
- 8) Maka didapat pesanan mana saja yang harus diterima untuk mendapatkan keuntungan paling maksimal

```
max_Prof = 0
for i = 0 to n-1
  if max_Prof < temp_Prof[i]
    max_Prof = temp_Prof[i]
```

2. Prosedur yang memunculkan nama pesanan-pesanan yang diterima untuk mendapatkan keuntungan paling maksimal

```
Procedure cariPesanan(Pesanan, temp_Prof, max_Prof) :
for i = n-1 down to 0 and max_Prof > 0
  if max_Prof is equal to temp_Prof[i]
    output(p[i].name)
    max_Prof = max_Prof - p[i].profit
  endif
endfor
```

Berikut hasil yang didapat dari implementasi untuk penyelesaian masalah ini

```
C:\Users\PANGKOOPSAU1\Desktop>stima.py
Masukan banyaknya pesanan :5
Masukan Nama Pesanan :A
Masukan Start Time :2/4/2019
Masukan End Time :5/4/2019
Masukan Profit (Rp) :12000000

Masukan Nama Pesanan :B
Masukan Start Time :6/4/2019
End Time :7/4/2019
Profit (Rp) :8000000

Masukan Nama Pesanan :C
Masukan Start Time :7/4/2019
Masukan End Time :9/4/2019
Masukan Profit (Rp) :4000000

Masukan Nama Pesanan :D
Masukan Start Time :1/4/2019
Masukan End Time :3/4/2019
Profit (Rp) :10000000

Masukan Nama Pesanan :E
Masukan Start Time :5/4/2019
Masukan End Time :8/4/2019
Profit (Rp) :22000000

Masukan Nama Pesanan :F
Start Time :4/4/2019
End Time :6/4/2019
Profit (Rp) :10000000

Didapat keuntungan maksimum yaitu : 34000000

Pesanan yang perlu diambil untuk mendapatkan hasil maksimum :
Nama pesanan : A
Nama pesanan : E
```

Gambar 3. Hasil Implementasi

IV. KESIMPULAN

Berdasarkan hasil uji coba dari implementasi algoritma tersebut dan analisis, dapat disimpulkan bahwa optimasi keuntungan produksi, selain menggunakan algoritma *greedy*,

Berikut notasi algoritmik yang saya gunakan dalam penyelesaian menggunakan program dinamis :

1. Prosedur penjadwalan menggunakan program dinamis.

```
Procedure PD_scheduling(Pesanan p)
for i = 2 to n
  for j = 1 to i-1
    if p[j].end_time <= p[i].start_time
      if temp_Prof[j] + p[i].Profit > temp_Prof[i]
        temp_Prof[i] = temp_Prof[j] + p[i].Profit
      endif
    endif
  endfor
endfor
dilanjutkan setelah ini
```

bisa juga diselesaikan melalui pendekatan program dinamis. Semoga program ini dapat digunakan dengan tujuan memajukan produksi produk-produk dalam negeri dan memajukan ekonomi bangsa.

UCAPAN TERIMA KASIH

Pertama-tama penulis mengucapkan syukur kepada Allah SWT karena berkat rahmat dan pertolongan-Nya penulis dapat menyelesaikan perkuliahan Strategi Algoritma beserta tugas tugasnya dengan lancar pada semester ini.

Kemudian penulis juga berterima kasih kepada dosen-dosen pengajar mata kuliah IF2211 Strategi Algoritma, Drs. Nur Ulfa Maulidevi, Dr. Masayu Leylia Khodra ST., MT., dan Dr. Ir. Rinaldi Munir, M.T. Atas ilmu, saran dan dukungan yang telah dosen-dosen berikan kepada penulis.

Penulis juga berterima kasih kepada keluarga yang selalu memberikan dukungan dan motivasi untuk tetap semangat melalui perkuliahan ini

Tidak lupa penulis mengucapkan terima kasih kepada teman-teman yang telah menemani penulis baik dalam mengerjakan tugas maupun dalam belajar dan diskusi bersama. Alhamdulillah makalah ini selesai berkat dukungan semuanya.

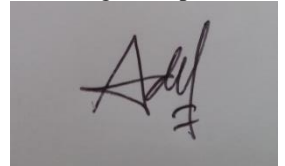
REFERENSI

- [1] Munir, Rinaldi. Diktat Kuliah IF2211 Strategi Algoritma. Program Studi Teknik Informatika ITB. 2018.
- [2] https://www.academia.edu/8912194/Pengantar_Ekonomi_teor_i_produk_si_biaya_dan_keuntungan, diakses pada 25 April 2019
- [3] <https://ditotomarunoto.wordpress.com/2014/04/27/produksi/>, diakses pada 25 April 2019
- [4] <https://www.maxmanroe.com/pengertian-konveksi.html>., diakses pada 25 April 2019
- [5] <https://www.geeksforgeeks.org/weighted-job-scheduling/>, diakses pada 25 April 2019

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 26 April 2019



Abdurrahman Adni 13517117