**LAPORAN PRAKTIKUM ALGORITMA DAN PEMOGRAMAN**

**MODUL KE – 10**

Perulangan di Dalam Perulangan(Nested Loop)

****

**Oleh :**

**Nama : Emanuel Matroni Ndiwa**

**NIM : 235314103**

**Kelas : DP**

**PROGRAM STUDI INFORMATIKA**

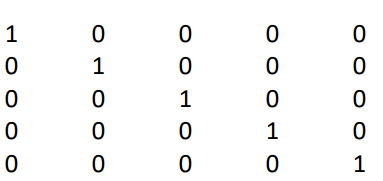
**FAKULTAS SAINS DAN TEKNOLOGI**

**UNIVERSITAS SANATA DHARMA**

**SEMESTER GASAL 2023/2024**

1. Tujuan
2. Mahasiswa menguasai penggunaan perintah loop bersarang untuk membuat program yang menuntut terjadinya perulangan bersarang/bertingkat.
3. Mahasiswa mampu menyelesaikan masalah konkrit sederhana memakai perintah perulangan bersarang.
4. Pelaksanaan praktikum
5. Soal nomor 1
6. Soal

Buat program yang dapat menampilkan matrik identitas N x N yakni N baris dan N kolom di mana N dimasukkan lewat keyboard. Matrik identitas adalah matrik yang semua elemennya berupa bilangan nol (0) kecuali elemen diagonal utama yang bernilai 1. Dengan kata lain jika aij adalah elemen baris ke i dan kolom ke j matrik maka aij = 0 untuk i ≠ j dan aij = 0 bila i = j. Untuk matrik 5 x 5 maka bentuknya adalah :



1. Pseudocode

MENCARI ELEMEN MATRIKS

Cetak masukan nilai N

For i i < N i++

For j j<N j++

If i = = j

Nilai=1

Else

Nilai = 0

Cetak Nilai

1. Kode program

package laporan\_dp\_10;

import java.util.Scanner;

public class SOAL\_01 {

public static void main(String[] args) {

Scanner user = new Scanner(System.in);

// meminta user memasukana nilai N untuk mencari elemen matriks N\*N

System.out.print("Masukan Nilai N untuk matriks identitas(N\*N)\t:");

int N = user.nextInt();

// kita memakai dua perulangan untuk memeriksa dan membuat nilai identitas 1 matriks elemennya

// kita membuat perulangan paling luar untuk melakukan pemeriksaan

// apakah kuanternya lebih kecil dari nilai n yang di masukan user

// jika lebih kecil maka prosesnya akan berlanjut ke baris perulangan berikutnya

// jika tidak maka proses akan langsung lompat ke baris paling luar yaitu printLN

for (int i = 0; i < N; i++) {

// perulangan ke dua akan di jalankan jika perulangan pertama di penuhi

// jika perulangan di penuhi dan perulangan ke dua di penuhi maka program akan lanjut ke baris berikutnya

for (int j = 0; j < N; j++) {

// untuk memeriksa apakah i = j atau i != j maka kita membutuhkan if-else statement

// untuk memeriksa nilai kebenaran

// jika benar maka print 1 jika salah 0

int nilai;

if (i == j) {

nilai = 1;

} else {

nilai = 0;

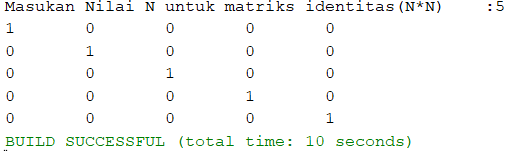
}

System.out.print(nilai + "\t");}

System.out.println("");

}

1. Capture output



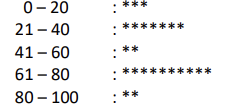
1. Analisis hasil

Dalam program ini kita di minta untuk membuat program mencari elemen matriks N\*N nilai N di masukan oleh user.

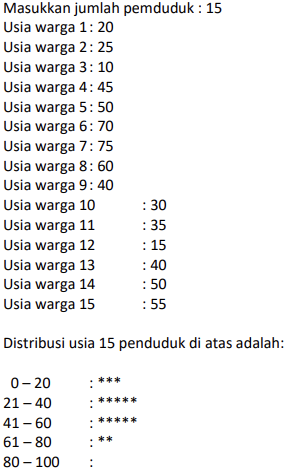
Untuk membuat program dengan perintah seperti ini beberapa langkah ini perlu kita lakukan unutk membuat programnya:

1. Memanggil library scanner java karena di program kita meminta inputan dari user.
2. Meminta user memasukan nilai N untuk mencari elemen matriks N\*N.
3. Dalam peogram ini kita membutuhkan perulangan bersarang yang di mana perulangan paling luar dan perulangan paling dalam memiliki perannya masing-masing.
4. Untuk kondisi for yang paling luar kita akan membuatvariabel kuanter kemudian dengan nilai awal 0 dan akan selalu lebih kecil dari nilai N.
5. Sama halnya yang akan kita lakukan pada kondisi perulangan paling dalamnya hanya kita membuat nama variabel yang berbeda.
6. Kemudian kita membutuhkan kondisi percabangan if dan else untuk memeriksa kondisi variabel kuanter paling luar dan paling dalam pada saat menjalankan proses perulangan ,apakah variabel kuantor paling luar = = kuantor paling dalam jika sama maka akan bernilai 1 jika salah maka bernilai 0.kita juga bisa menggunakan operator ternary jika nilai for luar= = dalam maka akan bernilai benar ketika benar maka akan mencetak angka 1 dan jika salah mencetak nilai 0.
7. Dan paling akhir kita akan mencetak nilai dari hasil proses di dalam percabangan.
8. Soal nomor 2
9. Soal

Buat program yang dapat menampilkan diagram batang mendatar dari data usia penduduk yang berjumlah N di mana usia dan N dimasukkan lewat keyboard. Data penduduk tersebut disajikan memakai format berikut:



Di dalam diagram di atas, satu bintang mewakili satu penduduk yang berusia pada rentang data tersebut. Contoh input dan output program Anda adalah sbb:



1. Pseudocode

MENAMPILKAN DIAGRAM BATANG DARI RENTANGAN USIA

Cetak masukan jumlah penduduk

For i i≤jumlah penduduk i++

if (usia <= 20)

Hitung0\_20++

else if (usia <= 40)

Hitung21\_40++

else if (usia <= 60)

Hitung41\_60++

else if (usia <= 80)

Hitung61\_80++;

else

hitung81\_100++

cetak diagram batang

for j j≤ Hitung0\_20 j++

cetak \*

for j j≤ Hitung21\_41 j++

cetak \*

for j j≤ Hitung41\_60++

cetak \*

for j j≤ Hitung61\_80j++

cetak \*

for j j≤ Hitung81\_100 j++

cetak \*

1. Kode program

package laporan\_dp\_10;

import java.util.Scanner;

public class SOAL\_02 {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

// Meminta input jumlah penduduk

System.out.print("Masukkan jumlah penduduk: ");

int jumlah = scanner.nextInt();

// Menggunakan variabel untuk menyimpan jumlah penduduk pada masing-masing rentang usia

int Hitung0\_20 = 0, Hitung21\_40 = 0, Hitung41\_60 = 0, Hitung61\_80 = 0, hitung81\_100 = 0;

// Memasukkan usia penduduk

for (int i = 1; i <= jumlah; i++) {

System.out.print("Usia warga " + i + "\t: ");

int usia = scanner.nextInt();

// Menghitung jumlah rentangan distribusi usia

//agar kita dapat mengelompokan jenis rentangan sekaligus menghitungnya

//kita membutuhkan percabangan if else

if (usia <= 20) {

Hitung0\_20++;

} else if (usia <= 40) {

Hitung21\_40++;

} else if (usia <= 60) {

Hitung41\_60++;

} else if (usia <= 80) {

Hitung61\_80++;

} else {

hitung81\_100++;

}

}

// Menampilkan diagram batang mendatar

//diagram batang ini di peroleh melalui pengelompokan serta

//menjumlahkan jenisnya di percabangan if else

//satu bintang mewakili satu penduduk yang berusia pada rentangan data tersebut

System.out.println("Distribusi usia " + jumlah + " penduduk di atas adalah\t:");

// Diagram batang untuk rentang usia 0-20

System.out.print("0 - 20 : ");

//kondisi perulangan

//yang di mana kuanter dimulai dari nol

//serta kuanter akan melakukan pengulangan dan menjumlahkan bintang selama lebih kecil

//dari total pengelompokan rentangan yang sudah di buat pada percabangan

for (int j = 0; j < Hitung0\_20; j++) {

System.out.print("\*");

}

//jika kuanter sudah melebihi batas ketentuan rentangan maka dia akan turun ke baris berikutnya

System.out.println();

// Diagram batang untuk rentang usia 21-40

System.out.print("21 - 40 : ");

//kondisi kuanter untuk rentangan 21-40

//kuanter akan melaksanakan tugas pengulangan dan menjumlahkan bintang

//selama kuanter nilainya lebih kecil dari jumlah rentangan usia 21-40

//yang sudah di jumlahkan total kelompoknya di percabangan

for (int j = 0; j < Hitung21\_40; j++) {

System.out.print("\*");

}

//jika kuanter sudah melebihi batas ketentuan rentangan maka dia akan turun ke baris berikutnya

System.out.println();

// Diagram batang untuk rentang usia 41-60

System.out.print("41 - 60 : ");

//kondisi kuanter untuk rentangan 21-40

//kuanter akan melaksanakan tugas pengulangan dan menjumlahkan bintang

//selama kuanter nilainya lebih kecil dari jumlah rentangan usia 41-60

//yang sudah di jumlahkan total kelompoknya di percabangan

for (int j = 0; j < Hitung41\_60; j++) {

System.out.print("\*");

}

//jika kuanter sudah melebihi batas ketentuan rentangan maka dia akan turun ke baris berikutnya

System.out.println();

// Diagram batang untuk rentang usia 61-80

System.out.print("61 - 80 : ");

//kondisi kuanter untuk rentangan 61-80

//kuanter akan melaksanakan tugas pengulangan dan menjumlahkan bintang

//selama kuanter nilainya lebih kecil dari jumlah rentangan usia 41-60

//yang sudah di jumlahkan total kelompoknya di percabangan

for (int j = 0; j < Hitung61\_80; j++) {

System.out.print("\*");

}

//jika kuanter sudah melebihi batas ketentuan rentangan maka dia akan turun ke baris berikutnya

System.out.println();

// Diagram batang untuk rentang usia 81-100

System.out.print("81 - 100: ");

//kondisi kuanter untuk rentangan 61-80

//kuanter akan melaksanakan tugas pengulangan dan menjumlahkan bintang

//selama kuanter nilainya lebih kecil dari jumlah rentangan usia 81-100

//yang sudah di jumlahkan total kelompoknya di percabangan

for (int j = 0; j < hitung81\_100; j++) {

System.out.print("\*");

}

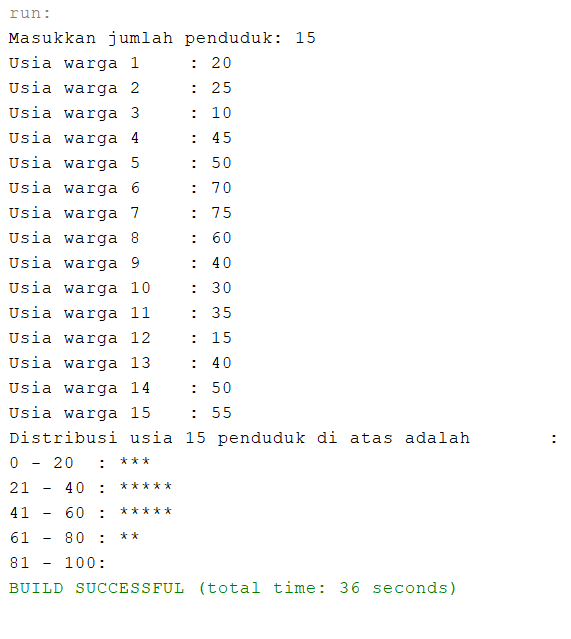
//jika kuanter sudah melebihi batas ketentuan rentangan maka dia akan turun ke baris berikutnya

System.out.println();

}

}

1. Capture output



1. Analisis hasil

Dari soal di atas kita di minta untuk membuat program diagram batang dari kelompok rentangan usia yang di masukan user.berikut langkah-langkah yang perlu diperhatikan:

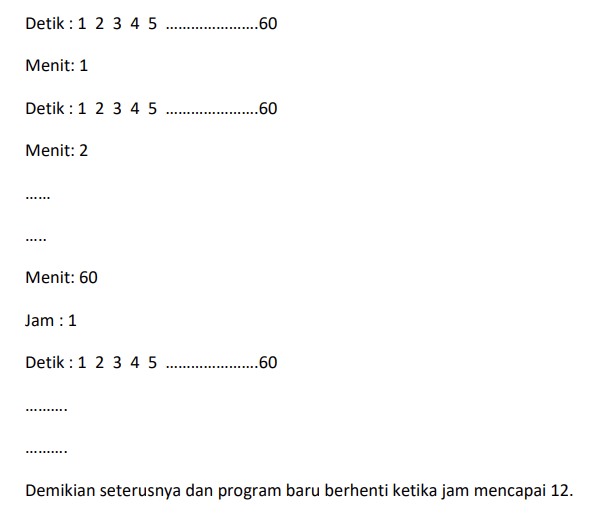
1. Memanggil library scanner java untuk meminta input dari user
2. Meminta user input jumlah penduduk
3. Membuat variabel untuk menyimpan jumlah penduduk berdasarkan rentangan usia masing-masing.dari variabel yang sudah kita buat kita perlu memberi nilai awal atau kita inisiasi.perlunya inisiasi adalah ketika kita membuat persamaan penjumlahan variabel ,variabel yang kita gunakan sudah memiliki nilai jika belum memiliki nilai awal maka variabel ini tidak bisa kita jumlahkan dan dengan memberikan nilai awal membantu kita memerikan hasil yang akurat sesuai logika program yang kita buat.
4. Kemudian kita membuat perulangan for untuk meminta user memasukan usia penduduk sebanyak jumlah penduduk yang yang user masukan tadi.
5. Dari usia penduduk yang di masukan kita perlu langsung mengelompokkan sesuai rentangan usia.untuk bisa mengelompokkanya kita membutuhkan percabangan if else dan di dalam percabangan kita langsung menjumlahkan rentangan usia tersebut. contohnya: if (usia≤20) kelompok usia 0-20++
6. Lalu kita akan menampilkan diagram batang dari hasil pengelompokkan usia tersebut.diagram batangnya mengunakan simbol bintang,satu bintang mewakili usia satu penduduk berdasarkan rentangan yang sudah di kelompokan.
7. Untuk menampilkan diagram batang ini kita memerlukan looping for pada setiap rentangan usia.kuanter setiap rentangan variabelnya memiliki nilai awal 0 dan variabel kuanter lebih kecil dari rentangan usia

Contohnya : for (int j = 0; j < Hitung61\_80; j++) {

System.out.print("\*");

1. Soal nomor 3
2. Soal

Buat program untuk menampilkan counter waktu seperti berikut



1. Pseudocode

MEMBUAT COUNTER WAKTU

For jam=1 jam≤12 jam++

For menit =1 menit ≤60 menit++

For detik=1 detik≤60 detik ++

Cetak detik

Cetak menit

1. Kode program

package laporan\_dp\_10;

public class SOAL\_03 {

public static void main(String[] args) {

//membuat looping for paling luar yaitu jam karena untuk memenuhi 1 jam

//membutuhkan perhitungan menit dan detik sehingga looping untuk jam di simpan di bagian luar

//karena di minta jam akan berhenti ketika genap 12 maka kuanternya kita buat akan <=12 dan di mulai dari 1

for (int jam = 1; jam <= 12; jam++) {

//sebelum jam ada menit untuk genap 1 jam maka di butuhkan waktu 60 menit

//untuk itu di bawah perulangan jam kita buat perulangan menit

//dan selama menit belum mencapai batasannya 60 maka dia akan melakukan perhitungan terus

//untuk memenuhi perhitungan itu kita membuat kuanter untuk menit dengan variabelnya di mulai angka 1

//dan menit <=60 karena 1 jam = 60 menit maka kita membuat perulangan dan menit akan dijumlahkan terus hingga 60

for (int menit = 1; menit <= 60; menit++) {

//kita mencetak tulisan detik di bagian sini atau sebelum looping untuk detik

//jika kita mencetaknnya di bawah setelah perintah looping maka tulisan detik ini

//akan di cetak terus menerus sebanyak batasan kuanter ,sehingga tulisan detiknya akan sebanyak jumlah perhitungan detik

System.out.print("detik\t:");

//kita membuat perulangan detik di paling dalam

//dan untuk memenuhi 1 menit di butuhkan waktu 60 detik

//untuk kondisi kuanternya masih sama seperti kondisi kuantor menit hanya kita merubah nama variabelnya

for (int detik = 1; detik <= 60; detik++) {

//kemudian kita mencetak nilai detik dari setiap perhitungan

System.out.printf(detik + " ");

}

System.out.println("");

//mencetak nilai menit di bagian blok sini

//karena kita akan mencetak nilai menit setelah detik mencapai angka 60 maka cetak menit=1

//jika kita mencetka nilai menit langsung di blok loopingnya maka nilai menit dan detik di cetak bersamaan

System.out.println("Menit: " + menit);

}

//mencetak nilai jam ketika detik dan menit sudah terpenuhi terlebih dahulu

System.out.println("Jam: " + jam);

}

}

}

1. Cature output



1. Analisis hasil

untuk membuat program counter waktu kita perlu memahami perhitungan waktu jam menit dan detik.untuk memenuhi 1jam di butuhkan perhitungan waktu menit selama 60 kali dan untuk memenuhi 1 menit dibutuhkan perhitungan detik sebanyak 60 kali.dari sini sudah kelihatan untuk posisi looping yang akan kita masukan kondisinya.untuk looping jam akan kita simpan di paling luar dan di tengah menit kemudian looping detik berada di paling dalam.

***Loop Jam***:Variabel jam digunakan sebagai loop untuk jam. Loop ini berjalan dari 1 hingga 12 (inklusif), sesuai dengan format jam pada umumnya.

Di dalam loop jam, terdapat loop menit dan loop detik.

***Loop Menit***:Variabel menit digunakan sebagai loop untuk menit. Loop ini berjalan dari 1 hingga 60 (inklusif), mengingat 1 jam memiliki 60 menit.

Di dalam loop menit, terdapat loop detik.

***Loop Detik*** :Variabel detik digunakan sebagai loop untuk detik. Loop ini juga berjalan dari 1 hingga 60 (inklusif), sesuai dengan jumlah detik dalam 1 menit.

Di dalam loop detik, nilai detik dicetak menggunakan System.out.printf(detik + " ").

***Output Tampilan Detik*** :Setiap kali loop detik dieksekusi, nilai detik dicetak dalam satu baris. Setelah mencetak nilai detik dari 1 hingga 60, baris baru ditambahkan (System.out.println("")) untuk memisahkan nilai detik antara satu menit dengan menit berikutnya.

***Output Tampilan Menit***:Setelah mencetak nilai detik untuk satu menit, dilakukan pencetakan nilai menit (System.out.println("Menit: " + menit)).

***Output Tampilan Jam***:Setelah loop detik dan loop menit selesai untuk satu jam, dilakukan pencetakan nilai jam (System.out.println("Jam: " + jam)).

1. Soal nomor 4
2. Soal

Dengan memakai perintah do …. while () tampilkan menu berikut di layer Menu: 1. Memasukkan data usia

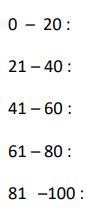
2. Rata-rata

3. Deviasi standar

4. Diagram batang

5. Keluar

Selanjutnya di dalam do -while berisi program-program untuk keperluan 5 menu di atas. Sebagai catatan, diagram batang yang anda buat menggunakan 5 kelompok data dan datanya harus dalam rentang 0 sampai 100. Artinya kelas datanya adalah



1. Pseudocode

MEMBUAT PILIHAN MENU DAN SETIAP MENU MENJALANKAN PERINTAH

Do

Cetak menu

Cetak 1.masukan data usia

Cetak 2.rata-rata

Cetak 3.deviasi standar

Cetak 4.diagram batang

Cetak 5.keluar

Switch pilihan

Case 1

Cetak masukan jumlah data ke

For a=1 a≤jumlah data a++

If jumlah data≥0 &&jumlah data≤20

Golongan 0-20++

Else if jumlah data ≥21&&jumlah data ≤40

Golongan 21-40++

Else if jumlah daata ≥41 && JUMLAH DATA ≤60

Golongan 41-60++

Else if julah data ≥61&&≤80

Golongan 61-80++

Else if jumlah data≥81&&≤100

Golongan 81-100++

Total +=jumlah data

Rata-rata=total?jumlah data

Case 2

Cetak rata-rata

Case 3

Cetak deviasi standar

Case 4

Cetak "Diagram Batang"

Cetak baris kosong

Cetak "0-20\t:"

Untuk setiap bintang dalam rentang A\_0

Cetak "\*"

Cetak baris kosong

Cetak "21-40\t:"

Untuk setiap bintang dalam rentang B\_21

Cetak "\*"

Cetak baris kosong

Cetak "41-60\t:"

Untuk setiap bintang dalam rentang C\_41

Cetak "\*"

Cetak baris kosong

Cetak "61-80\t:"

Untuk setiap bintang dalam rentang D\_61

Cetak "\*"

Cetak baris kosong

Cetak "80-100\t:"

Untuk setiap bintang dalam rentang E\_80

Cetak "\*"

Cetak baris kosong

Case 5

Cetak Anda keluar

Default

Cetak Anda salah input

While pilihan !=5

1. Kode program

package laporan\_dp\_10;

import java.util.Scanner;

public class SOAL\_04 {

public static void main(String[] args) {

Scanner user = new Scanner(System.in);

int pilihan;

int A\_0 = 0, B\_21 = 0, C\_41 = 0, D\_61 = 0, E\_80 = 0;

int rata = 0;

int total = 0;

int jumlah=0;

do {

System.out.println("Menu:");

System.out.println("1. Memasukkan data usia");

System.out.println("2. Rata-rata");

System.out.println("3. Deviasi standar");

System.out.println("4. Diagram batang");

System.out.println("5. Keluar");

System.out.print("Pilih menu (1-5): ");

pilihan = user.nextInt();

switch (pilihan) {

case 1:

System.out.print("masukan jumlah data \t:");

jumlah = user.nextInt();

for (int a = 1; a <= jumlah; a++) {

System.out.print("masukan jumlah data ke\t" + a+"\t:");

int jada = user.nextInt();

if (jada >= 0 && jada <= 20) {

A\_0++;

} else if (jada >= 21 && jada <= 40) {

B\_21++;

} else if (jada >= 41 && jada <= 60) {

C\_41++;

} else if (jada >= 61 && jada <= 80) {

D\_61++;

} else if (jada >= 80 && jada <= 100) {

E\_80++;

}

total += jada;

}

rata = total / jumlah;

break;

case 2:

System.out.println("Menghitung rata-rata\t:" + rata);

break;

case 3:

System.out.println("Menghitung deviasi standar");

System.out.println("Masukan Data Yang Sama Seperti Pilihan 1 Anda");

double totalSelisihKuadrat = 0;

for (int a = 1; a <= jumlah; a++) {

System.out.print("Masukkan data ke-" + a + ": ");

int data = user.nextInt();

double selisih = data - rata;

totalSelisihKuadrat += Math.pow(selisih, 2);

}

double deviasiStandar = Math.sqrt(totalSelisihKuadrat / jumlah);

System.out.println("Deviasi Standar: " + deviasiStandar);

break;

case 4:

System.out.println("Diagram Batang ");

System.out.println("");

System.out.print("0-20\t:");

for (int h = 0; h < A\_0; h++) {

System.out.printf("\*");

}

System.out.println("");

System.out.print("21-40\t:");

for (int j = 0; j < B\_21; j++) {

System.out.printf("\*");

}

System.out.println("");

System.out.print("41-60\t:");

for (int y = 0; y < C\_41; y++) {

System.out.printf("\*");

}

System.out.println("");

System.out.print("61-80\t:");

for (int u = 0; u < D\_61; u++) {

System.out.printf("\*");

}

System.out.println("");

System.out.print("80-100\t:");

for (int l = 0; l < E\_80; l++) {

System.out.printf("\*");

}

System.out.println("");

break;

case 5:

System.out.println("Anda Keluar +\_\_\_+ Terima kasih!");

break;

default:

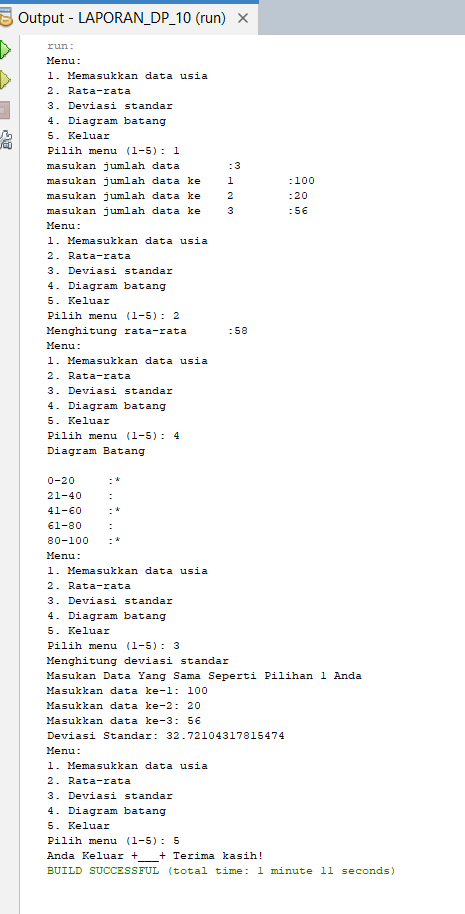
System.out.println("Anda salah memilih!!!! Silakan pilih 1-5.");

break;

}

} while (pilihan != 5);

1. Capture ouput



1. Analisis hasil

Unutuk membuat progam dengan output seperti pada soal ini kita punta bnyak piihan untuk membuat outputnya nammun kita perlu membuat pilihan output yang lebih efisien ubtuk itu berikut langkah-langkah membuat program seperti soal di atas :

Kita harus membuat terlebih dahulu variabel yang di perlukan di dalam program kita sseperti jenis rentangan,rata-rata,total dan jumlah.jika kita membuat variabel untuk mencari persamaan kita simpan di block percabangannya otomatis nanti ada yang eror.berikut alur programnya

Program memiliki menu utama dengan lima opsi:

1. Memasukkan data usia:

Pengguna diminta untuk memasukkan jumlah data.Selanjutnya, program meminta pengguna untuk memasukkan data usia sebanyak jumlah yang telah dimasukkan sebelumnya.Data usia dikelompokkan ke dalam rentang 0-20, 21-40, 41-60, 61-80, dan 80-100, dan jumlah setiap rentang dihitung.Total dari semua data usia juga dihitung.Rata-rata dihitung dengan membagi total oleh jumlah data.

2. Rata-rata:

Menampilkan nilai rata-rata dari data usia yang telah dimasukkan.

3. Deviasi Standar:

Meminta pengguna untuk memasukkan ulang data usia yang sama seperti pada menu 1.Menghitung deviasi standar dari data usia menggunakan rumus statistik.

4. Diagram Batang:

Menampilkan diagram batang yang merepresentasikan jumlah data usia pada masing-masing rentang.

5. Keluar:

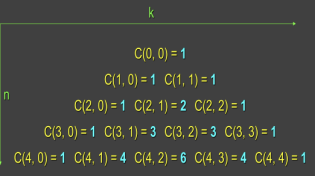
Keluar dari program.

1. Soal nomor 5
2. Soal

Segitiga Pascal adalah daftar bilangan bulat yang disusun berdasarkan jumlah dua bilangan sebelah kiri dan kanan dari deret sebelumnya seperti dalam gambar di bawah :



Angka-angka tersebut di atas juga dapat dihitung memakai rumus Kombinasi ( C(n,r) ) seperti dalam diagram berikut :



C(n,r) dapat dihitung memakai rumus n ! / [(n-r) ! r !] di mana n ! = 1 x 2 x 3 x … x n Sebagai contoh, C(4,2) = 4 ! / [(4-2) ! 2 !] = (1 x 2 x 3 x 4) / [ (1 x 2) (1 x 2)] = 6 Tugas Anda adalah menampilkan bilangan dari segitiga Pascal untuk baris atau n tertentu yang dimasukkan lewat keyboard. Jika nilai n adalah 3 maka program Anda menampilkan angka 1 3 3 1 dan angka-angka ini dapat dihitung masing-masing dengan C(3,0), C(3,1), C(3,2), C(3,3). Dengan kata lain, berdasarkan input besarnya n, program Anda harus menghitung dan menampilkan nilai dari C(n,0), C(n,1), C(n,2), …, C(n,(n-1)), C(n,n). Jika Anda sudah selesai mengerjakan tugas ini, program dapat Anda modifikasi sehingga mampu menghasilkan semua bilangan dalam segitiga Pascal mulai dari n = 0 sampai n bernilai tertentu yang dimasukkan lewat keyboard.

1. Pseudocode

BENTUK POHON BINTANG TETAPI PERHITUNGAN SEGITIGA PASCAL

Cetak masukan nilai N

For a a≤N a++

For b b ≤N-a b ++

Cetak spasi

For b b ≤a b++

Cetak nilai

Nilai = nilai \*a-b/(b+1)

1. Kode program

package laporan\_dp\_10;

import java.util.Scanner;

public class SOAL\_05 {

public static void main(String[] args) {

Scanner user = new Scanner(System.in);

//meminta user memasukan nilai N nilai yang ingin di cari oleh user untuk perhitungan

System.out.print("Masukan Nilai N segitiga pascal\t:");

int N = user.nextInt();

//untuk mencetak tinggi barisan sebanyak nilai N yang di masukan user maka perulangan

//untuk mencetak tinggi barisan di letakan paling luar karena setiap kali di periksapada perulangan ini

//maka akan cek dullu ke perulangan yang di dalamnya hingga selesai baru kembali ke perulangan ini lagi

for (int a = 0; a <= N; a++) {

for (int b = 0; b <= N - a; b++) {

//kemudian kita ingin membuat bentuknya pohon bintang maka di sini kita membutuhkan

//jumlah spasi ke kanan yang ingin kita gunakan berapa kali dan jumlah spasi pertama untuk menentukan bentuknya

System.out.print(" ");

//mencetak spasi ke kanan

}

//menginisiasikan dan membuat nilai awal variabel untuk membuat persamaan rumus kombinasi

int nilai = 1;

//perulangan yang paling dalam ini fungsinya untuk mencetak hasil persamaan rumus kombinasi tiap baris

//dan akan di jumlahkan sesuai banyaknya perulangan untuk mencari tinggi N

for (int b = 0; b <= a; b++) {

//membuat persamaan rumus kombinasi

//mencetak nilai atau angka hasil persamaan kombinasi

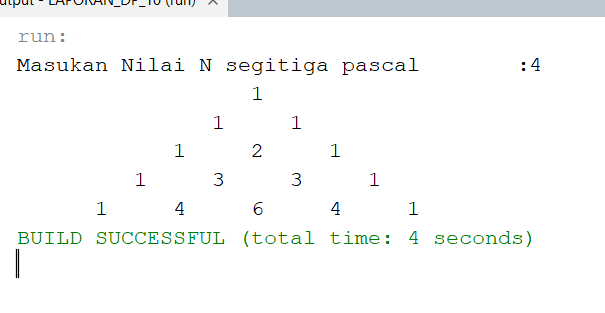
System.out.print(" " + nilai + " ");

nilai = nilai \* (a - b) / (b + 1);

}

System.out.println("");

1. Cature output



1. Analisis hasil

Ketika kita menemukan output untuk program seperti pada nomor lima,proses pembuatan loopingnya sama seperti untuk membuat looping pohon bintang tinggal kita menambahkan rumus kombinasi kedalam looping .unutk memasukan looping yang mana kita perlu memahami alur jalannya program terlebih dahulu.berikut ini alur programnya :

1. Input dari user

Kita meminta user memasukan jumlah nilai N untuk tinggi segitiga pascalnya

1. Perulangan pertama

Program menggunakan dua perulangan bersarang. Perulangan pertama (a) digunakan untuk mengatur tinggi barisan segitiga Pascal. Setiap perulangan ini mencetak satu baris dari segitiga.

1. Perulangan kedua

Perulangan kedua (b) digunakan untuk mengatur jumlah spasi dan mencetak nilai segitiga Pascal pada setiap baris.

1. Perhitungan dan mencetak

Program menggunakan rumus kombinasi untuk menghitung nilai pada setiap posisi (a,b) dan mencetak nilai.didalam proses ini juga membutuhkan looping untuk batasan proses penjumlahannya akan berhenti ketika nilainya sama atau lebih kecil dari jumlah tinggi segitiga yang dibutuhkan.

1. Daftar pustaka

Priyatama,Johanes Eka.Pengantar Bahasa Pemograman Jawa.jogja :universitas sanata Dharma.