



JURUSAN INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS SANATA DHARMA YOGYAKARTA

LAPORAN PRAKTIKUM ALGORITMA DAN PEMROGRAMAN
TOPIK : Praktikum Modul X Pengenalan Subprogram/Metode

Nama : ELVINA LORENZA PHANG

NIM : 215314103

Tugas No : 1

a. Uraian soal

Modifikasilah program latihan nomor 2 di atas sehingga program Anda dapat menghitung besarnya $C(n,r) = n! / [(n-r)! r!]$ di mana besarnya n dan r dimasukkan lewat keyboard serta memakai syarat bahwa $r \leq n$.

b. Rancangan program

Output

---Menghitung Besar $C(n,r)$ ---

Masukkan besar nilai n : 6 (data dimasukkan user lewat keyboard)

Masukkan besar nilai r : 3 (data dimasukkan user lewat keyboard)

Nilai $C(n,r)$: 20

Proses

Kerjakan blok pernyataan

Menampilkan tulisan 'Masukkan besar nilai n :' dan meminta user memasukkan data besar nilai n lewat keyboard

Membaca data besar nilai n yang dimasukkan user lewat keyboard

Menampilkan tulisan 'Masukkan besar nilai r :' dan meminta user memasukkan data besar nilai r lewat keyboard

Membaca data besar nilai r yang dimasukkan user lewat keyboard

Program akan terus berulang selama r lebih besar dari n

Proses (lanjutan)

Memesan memori bernama Cnr dengan tipe int dan berisi nilai dari metode hasilCnr

Menampilkan tulisan 'Nilai C (n,r) :' lalu mencetak isi variabel Cnr

Membuat metode hasilCnr bertipe int, dan memiliki parameter int n, int r

Memesan memori bernama Cnr dengan tipe int

Menyimpan rumus dan hasil perhitungan $C(n,r)$ di memori bernama Cnr

Mengembalikan nilai metode Cnr yang harus bertipe int

Membuat metode faktorial

Memesan memori bernama hasFak yang bernilai 1, i dengan tipe int

Jika $n = 0$

Mengembalikan nilai 1 ke metode faktorial

Jika tidak

Untuk nilai awal $i=1$, syarat perulangan $i \leq n$, dan update isi variabel i

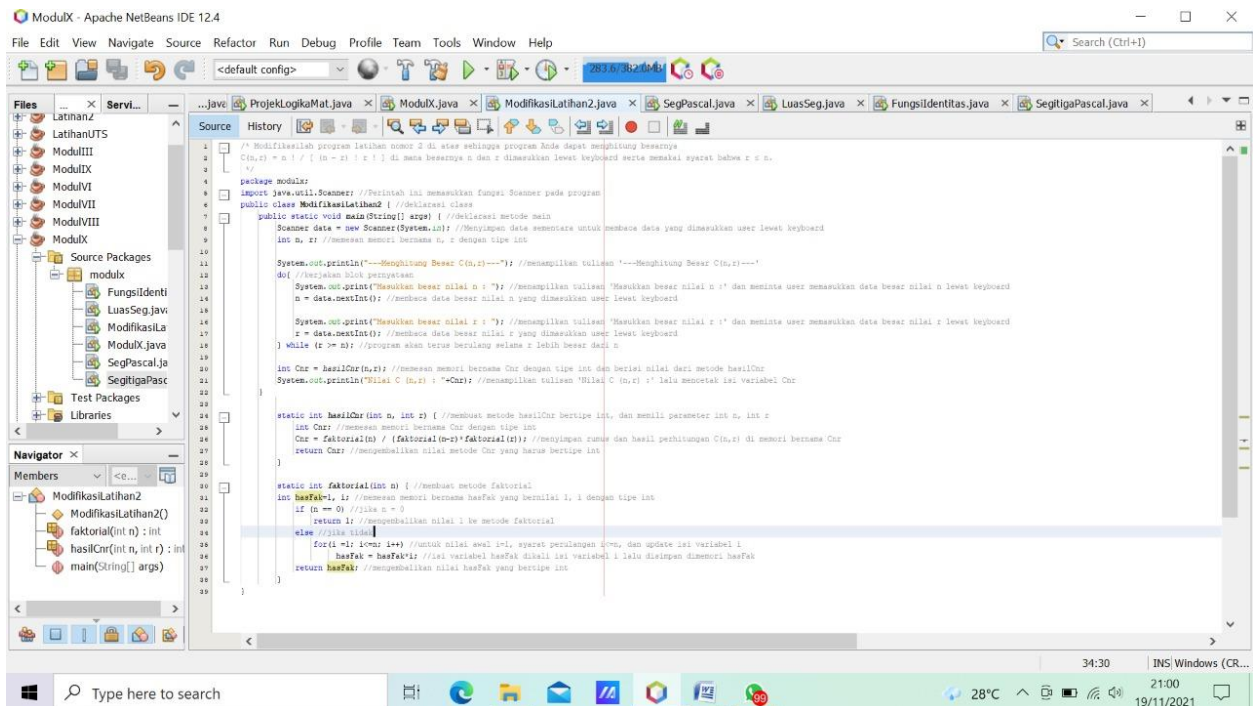
Isi variabel hasFak dikali isi variabel i lalu disimpan dimemori hasFak

Mengembalikan nilai hasFak yang bertipe int

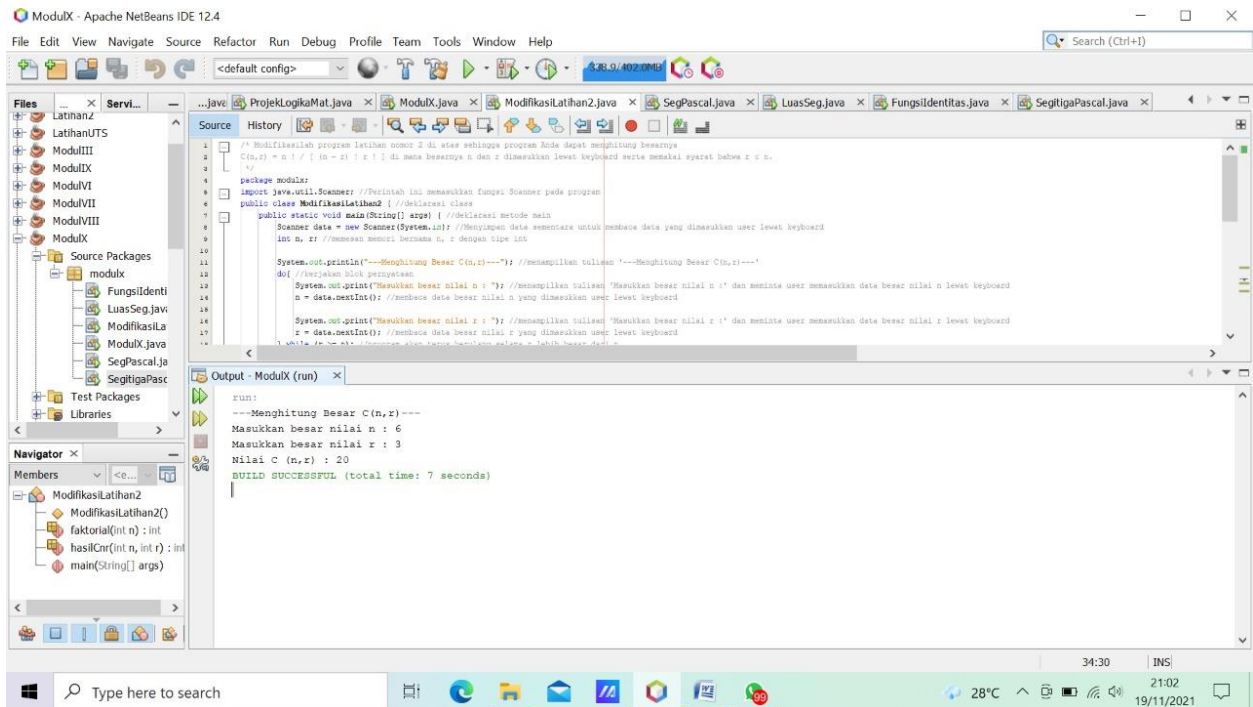
Input

Memesan memori bernama n, r dengan tipe int

Capture Program (didalam program berisi penjelasan setiap baris)



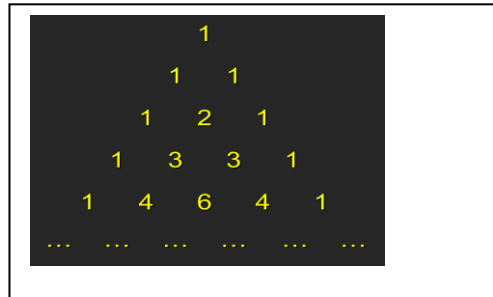
Capture Output



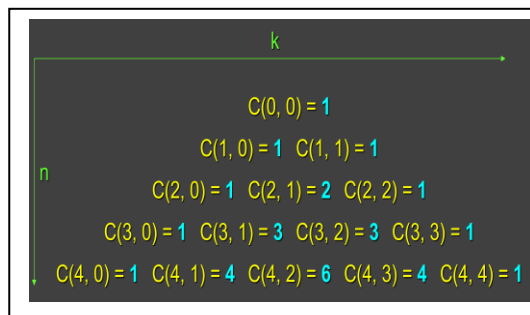
Tugas No : 2

a. Uraian 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



Ini adalah soal praktikum IX no 5. Modifikasilah program Anda memakai pendekatan modular sedemikian rupa program Anda memakai 2 buah metode yakni metode untuk menghitung $n!$ dan metode untuk menghitung $C(n,r)$.

b. Rancangan program

Output

Masukkan Nilai n : 10

```
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
1 5 10 10 5 1
1 6 15 20 15 6 1
1 7 21 35 35 21 7 1
1 8 28 56 70 56 28 8 1
1 9 36 84 126 126 84 36 9 1
1 10 45 120 210 252 210 120 45 10 1
```

Proses

Menampilkan tulisan 'Masukkan Nilai n : ' dan meminta user memasukkan data nilai n

Membaca data nilai n yang dimasukkan user lewat keyboard

Untuk nilai awal $i=0$, syarat perulangan $i \leq n$, dan update isi variabel i

 Untuk nilai awal $j=0$, syarat perulangan $j \leq i$, dan update isi variabel j

 Memesan memori bernama cij dengan tipe long dan di dalamnya berisi metode kombinasi(i,j)

 Menampilkan tulisan ' ' lalu menampilkan isi variabel cij dan menampilkan tulisan ' '

 Menampilkan " untuk membuat baris baru/enter

Membuat metode kombinasi bertipe long, dan memiliki parameter long n, long r

 Menyimpan rumus dan hasil perhitungan kombinasi di memori bernama komb dengan tipe long

 Mengembalikan nilai komb yang bertipe int

Proses (lanjutan)

Membuat metode faktorial bertipe long, dan memiliki parameter long k

Memesan memori bernama fak yang bernilai 1 dan bertipe long

Jika k = 0

Mengisi nilai 1 di memori bernama fak

Jika tidak

Untuk nilai awal l=1, syarat perulangan l<=k,

Isi variabel fak dikali 1 dan di simpan dimemori fak

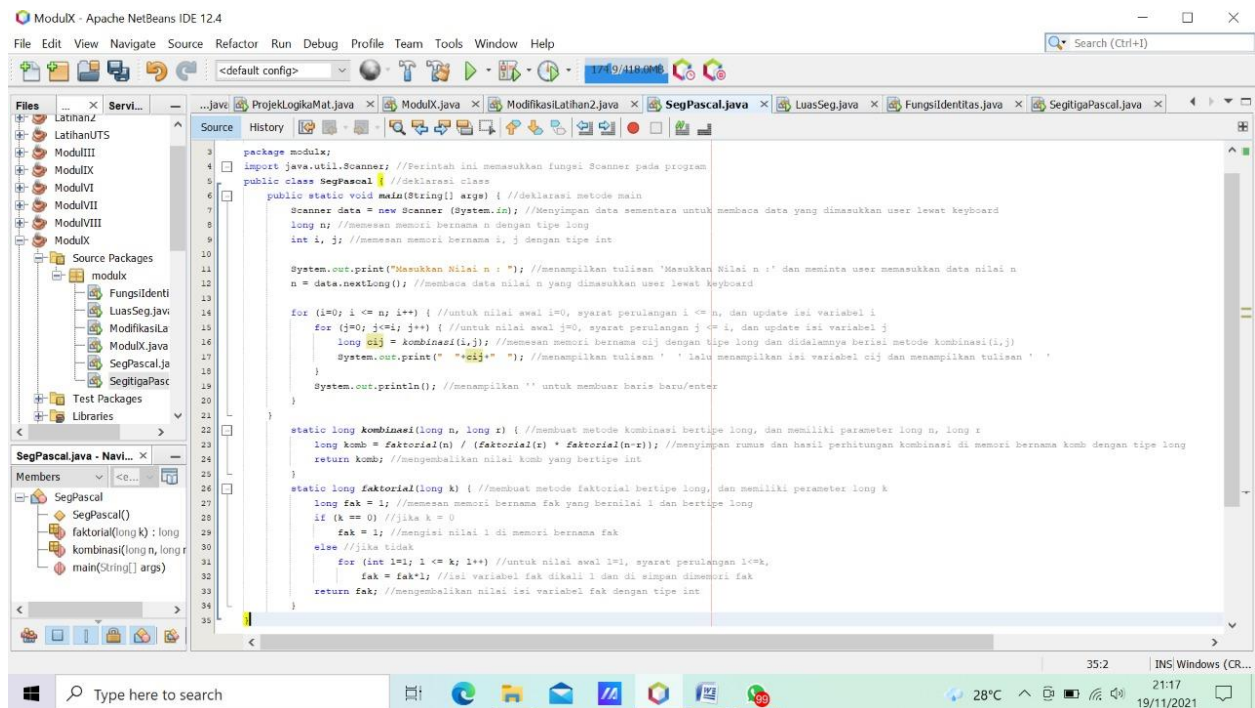
Mengembalikan nilai isi variabel fak dengan tipe int

Input

Memesan memori bernama n dengan tipe long

Memesan memori bernama i, j dengan tipe int

Capture Program (didalam program berisi penjelasan setiap baris)



```
1 package modul1;
2 import java.util.Scanner; //Perintah ini memasukkan fungsi Scanner pada program
3 public class SegPascal { //deklarasi class
4     public static void main(String[] args) { //deklarasi metode main
5         Scanner data = new Scanner(System.in); //Menyimpan data sementara untuk membaca data yang dimasukkan user lewat keyboard
6         long n; //memesan memori bernama n dengan tipe long
7         int i, j; //memesan memori bernama i, j dengan tipe int
8
9         System.out.print("Masukkan Nilai n : "); //menampilkan tulisan 'Masukkan Nilai n : ' dan meminta user memasukkan data nilai n
10        n = data.nextLong(); //membaca data nilai n yang dimasukkan user lewat keyboard
11
12        for (i=0; i <= n; i++) { //untuk nilai awal i=0, syarat perulangan i <= n, dan update isi variabel i
13            for (j=0; j <= i; j++) { //untuk nilai awal j=0, syarat perulangan j <= i, dan update isi variabel j
14                long komb = kombinasi(i,j); //memesan memori bernama i,j dengan tipe long dan didalamnya berisi metode kombinasi(i,j)
15                System.out.print(" " + komb + " "); //menampilkan tulisan ' ' lalu menampilkan isi variabel i,j dan menampilkan tulisan ' '
16            }
17            System.out.println(); //menampilkan '\n' untuk membuat baris baru/enter
18        }
19    }
20
21    static long kombinasi(long n, long x) { //membuat metode kombinasi bertipe long, dan memiliki parameter long n, long x
22        long komb = faktorial(n) / (faktorial(x) * faktorial(n-x)); //menyimpan rumus dan hasil perhitungan kombinasi di memori bernama komb dengan tipe long
23        return komb; //mengembalikan nilai komb yang bertipe int
24    }
25
26    static long faktorial(long k) { //membuat metode faktorial bertipe long, dan memiliki parameter long k
27        long fak = 1; //memesan memori bernama fak yang bernilai 1 dan bertipe long
28        if (k == 0) { //jika k = 0
29            fak = 1; //mengisi nilai 1 di memori bernama fak
30        } else { //jika tidak
31            for (int l=1; l <= k; l++) { //untuk nilai awal l=1, syarat perulangan l<=k,
32                fak = fak*l; //isi variabel fak dikali 1 dan di simpan dimemori fak
33            }
34            return fak; //mengembalikan nilai isi variabel fak dengan tipe int
35        }
36    }
37 }
```

Capture Output

The screenshot displays the Apache NetBeans IDE 12.4 interface. The main editor window shows the source code of a Java program named `SegPascal.java`. The code is as follows:

```
package module;  
import java.util.Scanner; //Perintah ini memasukkan fungsi Scanner pada program  
public class SegPascal { //deklarasi class  
    public static void main(String[] args) { //deklarasi metode main  
        Scanner data = new Scanner (System.in); //Menyimpan data sementara untuk membaca data yang dimasukkan user lewat keyboard  
        long n; //memesan memori bernama n dengan tipe long  
        int i, j; //memesan memori bernama i, j dengan tipe int  
  
        System.out.print("Masukkan Nilai n : "); //menampilkan tulisan 'Masukkan Nilai n : ' dan meminta user memasukkan data nilai n  
        n = data.nextLong(); //membaca data nilai n yang dimasukkan user lewat keyboard  
  
        for (i=0; i <= n; i++) { //untuk nilai awal i=0, syarat perulangan i <= n, dan update isi variabel i
```

The `Output - ModulX (run)` window shows the execution results:

```
run:  
Masukkan Nilai n : 10  
1  
1 1  
1 2 1  
1 3 3 1  
1 4 6 4 1  
1 5 10 10 5 1  
1 6 15 20 15 6 1  
1 7 21 35 35 21 7 1  
1 8 28 56 70 56 28 8 1  
1 9 36 84 126 126 84 36 9 1  
1 10 45 120 210 252 210 120 45 10 1  
BUILD SUCCESSFUL (total time: 4 seconds)
```

The bottom status bar indicates the system temperature is 28°C, the time is 17:26, and the date is 19/11/2021.

Tugas No : 3

a. Uraian soal

Buat program untuk menghitung luas segitiga berdasarkan koordinat 3 titik sudutnya yang dimasukkan lewat keyboard. Program Anda harus memakai beberapa metode berikut

- `bacaTitik()` // membaca absis dan ordinat suatu titik dan tanpa nilai balikan
- `hitungJarak(int x1, int y1, int x2, int y2)` // menghitung jarak antara 2 titik (x1,y1) dan (x2,y2) dengan nilai balikan (return) jarak dua titik
- `hitungKeliling(double sisiA, double sisiB, double sisiC)` //dengan nilai balikan keliling segitiga atau jumlah sisiA, sisiB dan sisiC.
- `hitungLuas()` // menghitung dan menampilkan luas segitiga memakai rumus yang sudah pernah diajarkan

b. Rancangan Program

Output

===Menghitung Luas Segitiga Berdasarkan Koordinat 3 Titik Sudut===

Masukkan titik koordinat x1 : 6 (data dimasukkan user lewat keyboard)

Masukkan titik koordinat x2 : 6 (data dimasukkan user lewat keyboard)

Masukkan titik koordinat x3 : 1 (data dimasukkan user lewat keyboard)

Masukkan titik koordinat y1 : -3 (data dimasukkan user lewat keyboard)

Masukkan titik koordinat y2 : 7 (data dimasukkan user lewat keyboard)

Masukkan titik koordinat y3 : 5 (data dimasukkan user lewat keyboard)

Jarak Antara dua titik (x1,y1 dan x2,y2) adalah : 10.0

Jarak Antara dua titik (x2,y2 dan x3,y3) adalah : 5.385164807134504

Jarak Antara dua titik (x1,y1 dan x3,y3) adalah : 9.433981132056603

Keliling nya adalah : 24.81914593919111

Luas nya adalah : 25.000000000000014

Proses

Variabel s1 diisi dengan isi metode hitungJarak(x1,x2,y1,y2)

Variabel s2 diisi dengan isi metode hitungJarak(x2,x3,y2,y3)

Variabel s3 diisi dengan isi metode hitungJarak(x1,x3,y1,y3)

Menampilkan tulisan 'Jarak Antara dua titik (x1,y1 dan x2,y2) adalah :' dan mencetak isi variabel s1

Menampilkan tulisan 'Jarak Antara dua titik (x2,y2 dan x3,y3) adalah :' dan mencetak isi variabel s2

Menampilkan tulisan 'Jarak Antara dua titik (x1,y1 dan x3,y3) adalah :' dan mencetak isi variabel s3

Menampilkan tulisan 'Keliling nya adalah :' dan mencetak isi metode hitungKeliling()

Menampilkan tulisan 'Luas nya adalah :' dan mencetak isi metode hitungLuas()

Membuat metode bacaData() yang bersifat static

Menampilkan tulisan '===Menghitung Luas Segitiga Berdasarkan Koordinat 3 Titik Sudut==='

Menyimpan data sementara untuk membaca data yang dimasukkan user lewat keyboard

Menampilkan tulisan 'Masukkan titik koordinat x1 : ' dan meminta user memasukkan data x1 lewat keyboard

Membaca data x1 yang dimasukkan user lewat keyboard

Menampilkan tulisan 'Masukkan titik koordinat x2 : ' dan meminta user memasukkan data x2 lewat keyboard

Membaca data x2 yang dimasukkan user lewat keyboard

Menampilkan tulisan 'Masukkan titik koordinat x3 : ' dan meminta user memasukkan data x3 lewat keyboard

Membaca data x3 yang dimasukkan user lewat keyboard

Menampilkan tulisan 'Masukkan titik koordinat y1 : ' dan meminta user memasukkan data y1 lewat keyboard

Membaca data y1 yang dimasukkan user lewat keyboard

Menampilkan tulisan 'Masukkan titik koordinat y2 : ' dan meminta user memasukkan data y2 lewat keyboard

Membaca data y2 yang dimasukkan user lewat keyboard

Proses (lanjutan)

Menampilkan tulisan 'Masukkan titik koordinat y3 : ' dan meminta user memasukkan data y3 lewat keyboard

Membaca data y3 yang dimasukkan user lewat keyboard

Membuat metode hitungJarak bertipe double, dan memiliki parameter double p1, double p2, double q1, double q2

Memesan memori bernama jarak dengan tipe double

Menyimpan rumus dan hasil perhitungan jarak antar dua titik dimemori bernama jarak

Mengembalikan nilai jarak yang bertipe double

Membuat metode hitungKeliling()

Memesan memori bernama kel dengan tipe double

Menyimpan rumus dan hasil perhitungan keliling di memori bernama kel

Mengembalikan nilai kel yang bertipe double

Membuat metode hitungLuas()

Memesan memori bernama luas yang berisi nilai 0 dengan tipe double

Memesan memori bernama s yang berisi rumus jarak untuk menghitung luas

Menyimpan rumus dan hasil perhitungan luas dimemori bernama luas

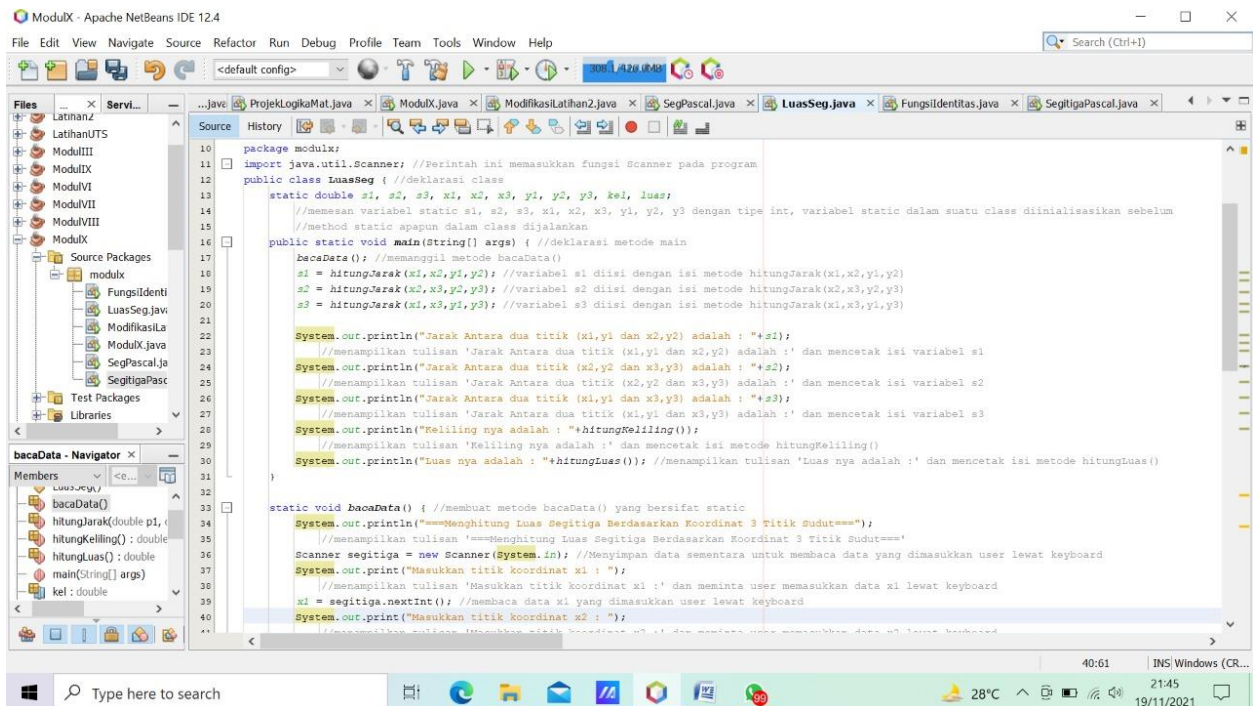
Mengembalikan nilai luas yang bertipe double

Input

Memesan variabel s1, s2, s3, x1, x2, x3, y1, y2, y3 dengan tipe int yang bersifat static

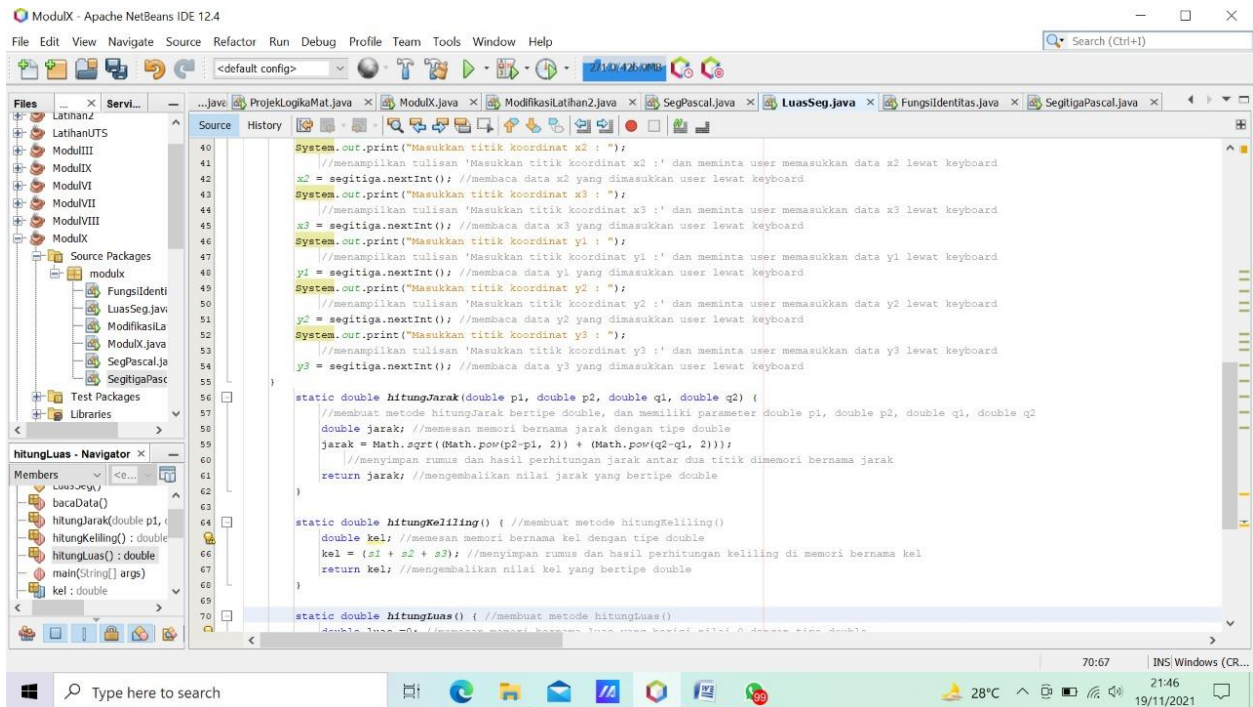
Memanggil metode bacaData()

Capture Program (didalam program berisi penjelasan setiap baris)



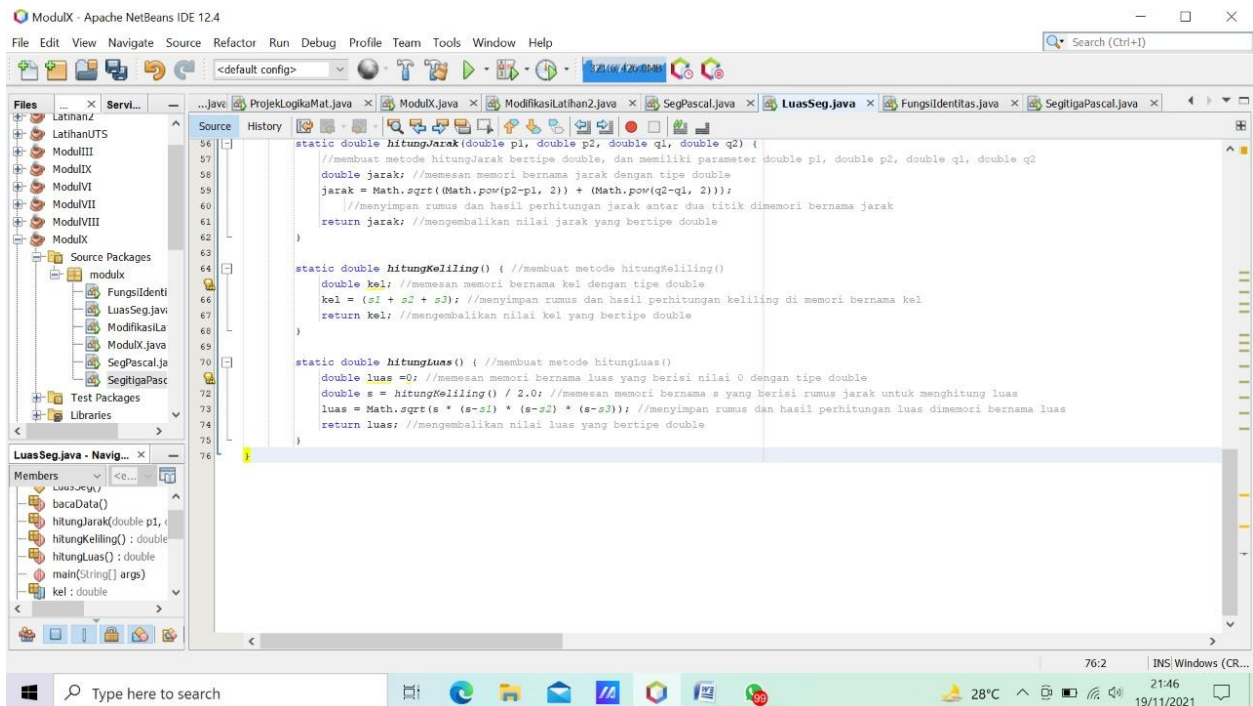
```
10 package modulx;
11 import java.util.Scanner; //Perintah ini memasukkan fungsi Scanner pada program
12 public class LuasSeg { //deklarasi class
13     static double s1, s2, s3, x1, x2, x3, y1, y2, y3, kel, luas;
14     //memesan variabel static s1, s2, s3, x1, x2, x3, y1, y2, y3 dengan tipe int, variabel static dalam suatu class diinisialisasikan sebelum
15     //method static apapun dalam class dijalankan
16     public static void main(String[] args) { //deklarasi metode main
17         bacaData(); //memanggil metode bacaData()
18         s1 = hitungJarak(x1, x2, y1, y2); //variabel s1 diisi dengan isi metode hitungJarak(x1, x2, y1, y2)
19         s2 = hitungJarak(x2, x3, y2, y3); //variabel s2 diisi dengan isi metode hitungJarak(x2, x3, y2, y3)
20         s3 = hitungJarak(x1, x3, y1, y3); //variabel s3 diisi dengan isi metode hitungJarak(x1, x3, y1, y3)
21
22         System.out.println("Jarak Antara dua titik (x1,y1 dan x2,y2) adalah : "+s1);
23         //menampilkan tulisan 'Jarak Antara dua titik (x1,y1 dan x2,y2) adalah : ' dan mencetak isi variabel s1
24         System.out.println("Jarak Antara dua titik (s2,y2 dan x3,y3) adalah : "+s2);
25         //menampilkan tulisan 'Jarak Antara dua titik (x2,y2 dan x3,y3) adalah : ' dan mencetak isi variabel s2
26         System.out.println("Jarak Antara dua titik (x1,y1 dan x3,y3) adalah : "+s3);
27         //menampilkan tulisan 'Jarak Antara dua titik (x1,y1 dan x3,y3) adalah : ' dan mencetak isi variabel s3
28         System.out.println("Keliling nya adalah : "+hitungKeliling());
29         //menampilkan tulisan 'Keliling nya adalah : ' dan mencetak isi metode hitungKeliling()
30         System.out.println("Luas nya adalah : "+hitungLuas()); //menampilkan tulisan 'Luas nya adalah : ' dan mencetak isi metode hitungLuas()
31     }
32
33     static void bacaData() { //membuat metode bacaData() yang bersifat static
34         System.out.println("====Menghitung Luas Segitiga Berdasarkan Koordinat 3 Titik Sudut====");
35         //menampilkan tulisan '====Menghitung Luas Segitiga Berdasarkan Koordinat 3 Titik Sudut===='
36         Scanner segitiga = new Scanner(System.in); //Menyimpan data sementara untuk membaca data yang dimasukkan user lewat keyboard
37         System.out.print("Masukkan titik koordinat x1 : ");
38         //menampilkan tulisan 'Masukkan titik koordinat x1 : ' dan meminta user memasukkan data x1 lewat keyboard
39         x1 = segitiga.nextInt(); //membaca data x1 yang dimasukkan user lewat keyboard
40         System.out.print("Masukkan titik koordinat x2 : ");
41         //menampilkan tulisan 'Masukkan titik koordinat x2 : ' dan meminta user memasukkan data x2 lewat keyboard
```

Capture Program Lanjutan (didalam program berisi penjelasan setiap baris)

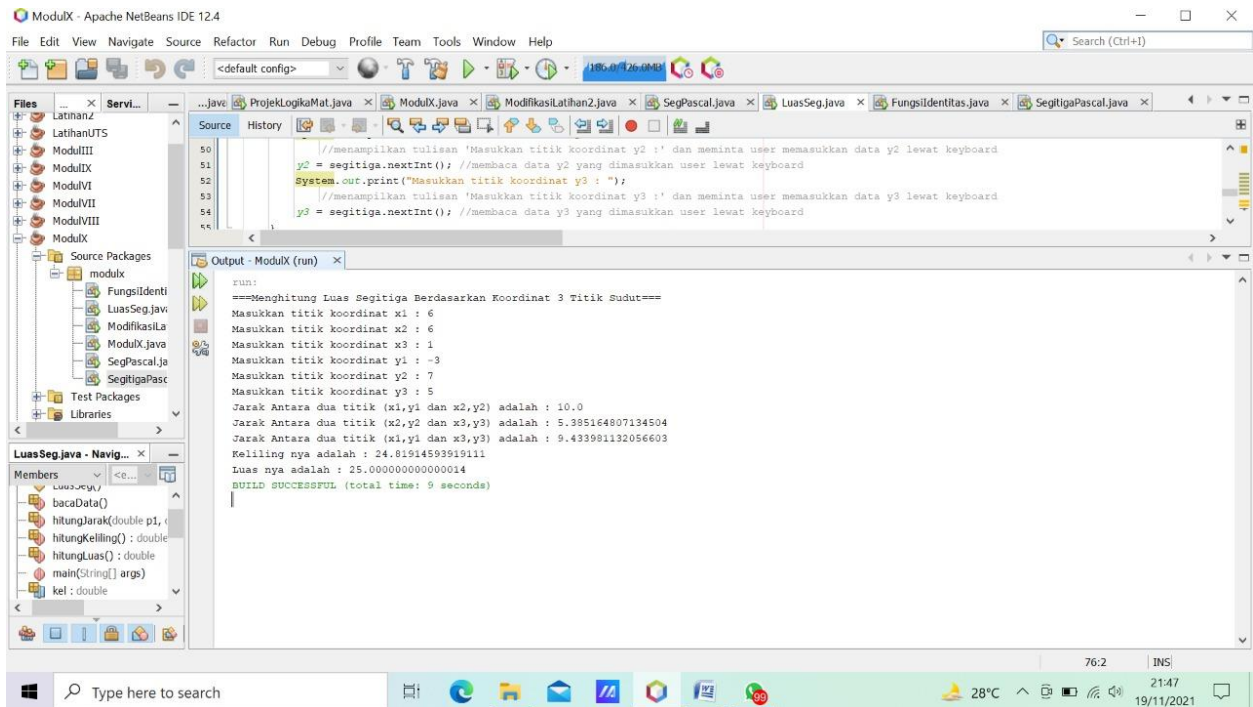


```
42         System.out.print("Masukkan titik koordinat x2 : ");
43         //menampilkan tulisan 'Masukkan titik koordinat x2 : ' dan meminta user memasukkan data x2 lewat keyboard
44         x2 = segitiga.nextInt(); //membaca data x2 yang dimasukkan user lewat keyboard
45         System.out.print("Masukkan titik koordinat x3 : ");
46         //menampilkan tulisan 'Masukkan titik koordinat x3 : ' dan meminta user memasukkan data x3 lewat keyboard
47         x3 = segitiga.nextInt(); //membaca data x3 yang dimasukkan user lewat keyboard
48         System.out.print("Masukkan titik koordinat y1 : ");
49         //menampilkan tulisan 'Masukkan titik koordinat y1 : ' dan meminta user memasukkan data y1 lewat keyboard
50         y1 = segitiga.nextInt(); //membaca data y1 yang dimasukkan user lewat keyboard
51         System.out.print("Masukkan titik koordinat y2 : ");
52         //menampilkan tulisan 'Masukkan titik koordinat y2 : ' dan meminta user memasukkan data y2 lewat keyboard
53         y2 = segitiga.nextInt(); //membaca data y2 yang dimasukkan user lewat keyboard
54         System.out.print("Masukkan titik koordinat y3 : ");
55         //menampilkan tulisan 'Masukkan titik koordinat y3 : ' dan meminta user memasukkan data y3 lewat keyboard
56         y3 = segitiga.nextInt(); //membaca data y3 yang dimasukkan user lewat keyboard
57
58     static double hitungJarak(double p1, double p2, double q1, double q2) {
59         //membuat metode hitungJarak bertipe double, dan memiliki parameter double p1, double p2, double q1, double q2
60         double jarak; //memesan memori bernama jarak dengan tipe double
61         jarak = Math.sqrt(Math.pow(p2-p1, 2) + (Math.pow(q2-q1, 2)));
62         //menyimpan rumus dan hasil perhitungan jarak antar dua titik di memori bernama jarak
63         return jarak; //mengembalikan nilai jarak yang bertipe double
64     }
65
66     static double hitungKeliling() { //membuat metode hitungKeliling()
67         double kel; //memesan memori bernama kel dengan tipe double
68         kel = (s1 + s2 + s3); //menyimpan rumus dan hasil perhitungan keliling di memori bernama kel
69         return kel; //mengembalikan nilai kel yang bertipe double
70     }
71
72     static double hitungLuas() { //membuat metode hitungLuas()
73         double luas; //memesan memori bernama luas dengan tipe double
```

Capture Program Lanjutan (didalam program berisi penjelasan setiap baris)



Capture Output



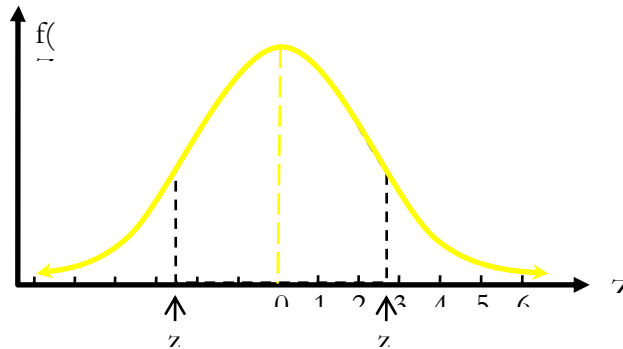
Tugas No : 4

a. Uraian soal

Fungsi idensitas/kepadatan normal baku yang berbentuk lonceng atau bel dapat dirumuskan memakai fungsi berikut ini.

$$f(Z) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}Z^2}, \text{dimana } -\infty < Z < +\infty$$

Di mana e adaah bilangan alam yang besarnya sama dengan 2,71828
Kalau digambar maka fungsi $f(z)$ di atas akan seperti ini



Gambar di atas juga bisa disajikan menjadi

Tugas Anda adalah membuat grafik/gambar fungsi normal di atas yang disederhanakan menjadi

*

**

* * * * *

* * * * *

* * * *

**

*

Caranya adalah:

- a. Buat metode yang menghitung nilai $f(z)$ tetapi karena $f(z)$ nilai maksimumnya hanya 1 maka buatlah metode tersebut menjadi menghitung nilai dari $n f(z)$ yang memberikan nilai balikan (return) int sebagai hasil pembulatan dari $n f(z)$. Dengan kata lain Anda harus membuat metode seperti berikut

```
static int normal (int n, double z) {
```

$$f = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}z^2}$$

```
    return (int) Math.round(n * f);  
}
```

- b. Buat metode untuk mencetak bintang sebanyak m buah yakni

```
static void bintang (int m) {
```

```
    for(int i = 1; i <= m; i++) System.out.print ("*");  
    System.out.println();  
}
```

- c. Buat class main yang memuat dua metode di atas serta memanggil metode bintang(m) di mana $m = \text{normal}(n, z)$ dan diulang untuk z mulai dari -4 sampai dengan 4 dengan pertambahan nilai 0.5 serta besarnya n dimasukkan lewat keyboard yang sebaiknya bernilai antara 30 sampai dengan 60. Dengan kata lain metode main Anda pada dasarnya berbentuk sbb

```
    baca n // membaca besarnya n (antara 10 – 20)  
    for (double z = -4; z <= 4; z = z+0.5) {  
        int jumBintang = normal(n,z);  
        bintang(jumBintang);  
    }
```

b. Rancangan Program

Output

---Grafik/Gambar Fungsi Identitas (Kepadatan Normal)---

Masukkan Nilai n : 45 (data dimasukkan user lewat keyboard)

```
*
**
*****
*****
*****
*****
*****
*****
*****
*****
*****
**
*
```

Input

Kerjakan blok pernyataan

Menampilkan tulisan 'Masukkan Nilai n : ' dan meminta user memasukkan data nilai n

Membaca data nilai n yang dimasukkan user lewat keyboard

Program akan terus berulang selama $n < 20$ atau $n > 100$

Proses (lanjutan)

Untuk nilai awal $z = -4$, syarat perulangan $z \leq 4$, dan update isi variabel z dengan menambahkan isi variabel z dengan 0.5

Memesan memori bernama `jumBintang` dengan tipe `int` dan berisi metode normal dengan parameter `n,z`

Memanggil metode bintang yang memiliki parameter berisi variabel `jumBintang`

Membuat metode normal bertipe `int`, dan memiliki parameter `int n, double z`

Memesan memori bernama `e`, `nilaiPangkat`, `f`, `akar`, `phi` dengan tipe `double`

Menyimpan nilai 2.71828 di memori bernama `e`

Menyimpan rumus dan hasil perhitungan nilai pangkat di memori bernama `nilaiPangkat`

Menyimpan nilai 3.14 di memori bernama `phi`

Menyimpan rumus dan hasil perhitungan fungsi identitas di memori bernama `f`

Mengembalikan nilai isi variabel `Math.round(n*f)` bertipe `int`

Membuat metode bintang, dan memiliki parameter `int m`

Untuk nilai awal $i=1$, syarat perulangan $i \leq m$, dan update isi variabel `i`

Menampilkan tulisan `'*`

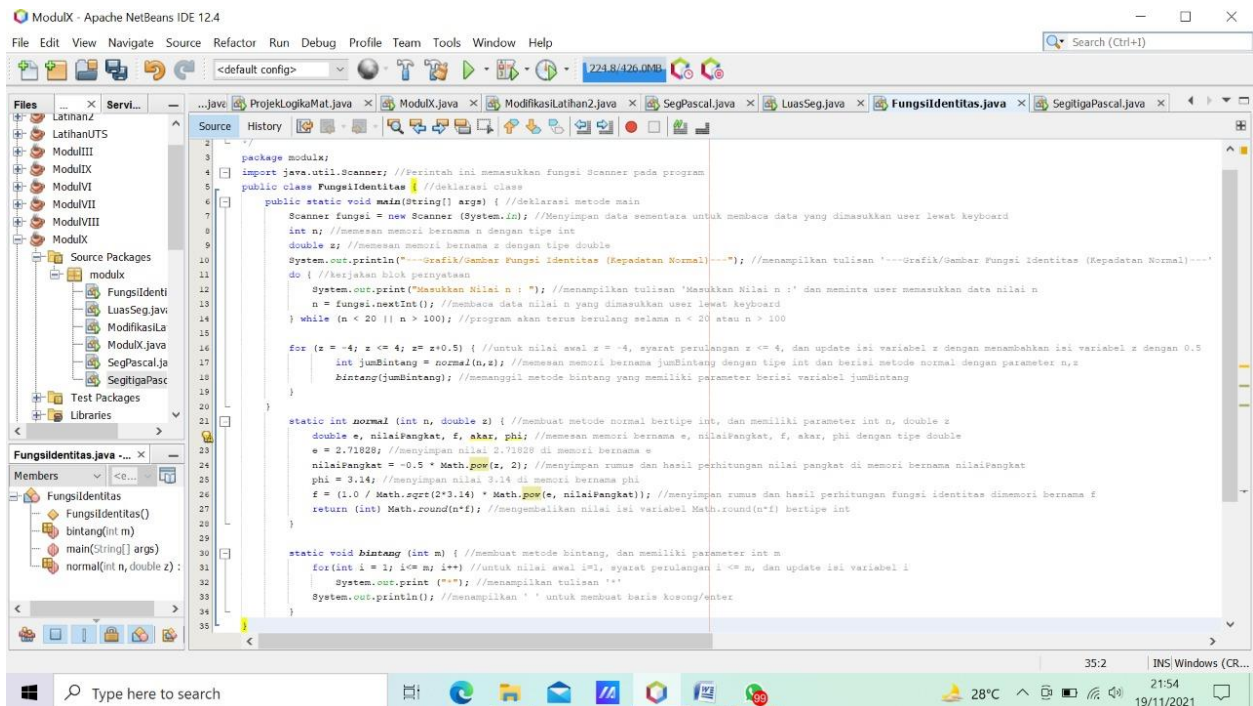
Menampilkan `' '` untuk membuat baris kosong/enter

Input

Memesan memori bernama `n` dengan tipe `int`

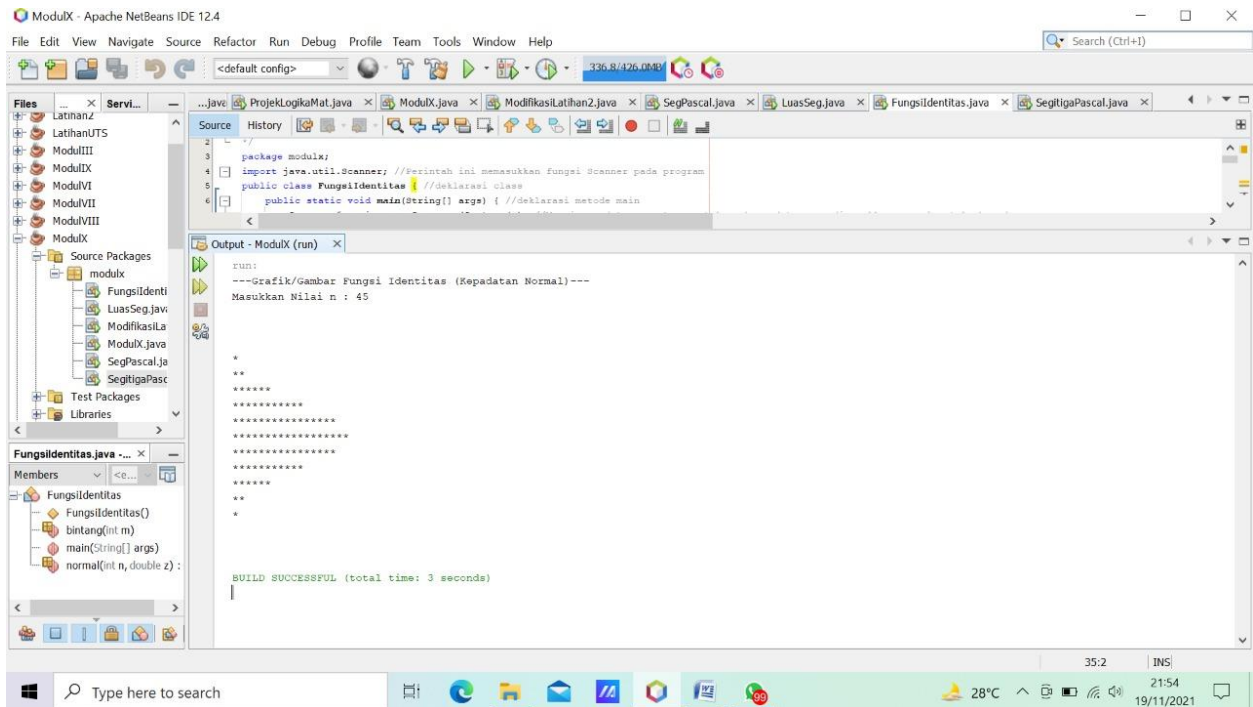
Memesan memori bernama `z` dengan tipe `double`

Capture Program (didalam program berisi penjelasan setiap baris)



```
1 package modules;
2 import java.util.Scanner; //Perintah ini memasukkan fungsi Scanner pada program
3
4 public class FungsilIdentitas { //deklarasi class
5
6     public static void main(String[] args) { //deklarasi metode main
7
8         Scanner fungsi = new Scanner(System.in); //Menyimpan data sementara untuk membaca data yang dimasukkan user lewat keyboard
9         int n; //memesan memori bernama n dengan tipe int
10        double z; //memesan memori bernama z dengan tipe double
11        System.out.println("---Grafik/Gambar Fungsi Identitas (Kepadatan Normal)---"); //menampilkan tulisan "---Grafik/Gambar Fungsi Identitas (Kepadatan Normal)---"
12        do { //kerjakan blok pernyataan
13            System.out.print("Masukkan Nilai n : "); //menampilkan tulisan "Masukkan Nilai n : " dan meminta user memasukkan data nilai n
14            n = fungsi.nextInt(); //membaca data nilai n yang dimasukkan user lewat keyboard
15            while (n < 20 || n > 100); //program akan terus berulang selama n < 20 atau n > 100
16
17            for (z = -4; z <= 4; z = z+0.5) { //untuk nilai awal z = -4, syarat perulangan z <= 4, dan update isi variabel z dengan menambahkan isi variabel z dengan 0.5
18                int jumBintang = normal(n,z); //memesan memori bernama jumBintang dengan tipe int dan berisi metode normal dengan parameter n,z
19                bintang(jumBintang); //memanggil metode bintang yang memiliki parameter berisi variabel jumBintang
20            }
21
22            static int normal (int n, double z) { //membuat metode normal bertipe int, dan memiliki parameter int n, double z
23                double e, nilaiFangkat, f, akar, phi; //memesan memori bernama e, nilaiFangkat, f, akar, phi dengan tipe double
24                e = 2.71828; //menyimpan nilai 2.71828 di memori bernama e
25                nilaiFangkat = -0.5 * Math.pow(z, 2); //menyimpan rumus dan hasil perhitungan nilai pangkat di memori bernama nilaiFangkat
26                phi = 3.14; //menyimpan nilai 3.14 di memori bernama phi
27                f = (1.0 / Math.sqrt(2*Math.PI)) * Math.pow(e, nilaiFangkat); //menyimpan rumus dan hasil perhitungan fungsi identitas di memori bernama f
28                return (int) Math.round(n*f); //mengembalikan nilai isi variabel Math.round(n*f) bertipe int
29            }
30
31            static void bintang (int m) { //membuat metode bintang, dan memiliki parameter int m
32                for(int i = 1; i <= m; i++) //untuk nilai awal i=1, syarat perulangan i <= m, dan update isi variabel i
33                    System.out.print (" "); //menampilkan tulisan " "
34                System.out.println(); //menampilkan " " untuk membuat baris kosong/enter
35            }
36        }
37    }
38 }
```

Capture Output



```
run:
---Grafik/Gambar Fungsi Identitas (Kepadatan Normal)---
Masukkan Nilai n : 45

**
*****
*****
*****
*****
*****
*****
**

BUILD SUCCESSFUL (total time: 3 seconds)
```