



# PERCABANGAN DAN PERULANGAN

ALGORITMA & PEMROGRAMAN I



Institut Teknologi Sumatera

# PRE TEST

- Apa beda struktur percabangan IF, IF-ELSE, IF-ELSE IF-ELSE, dan SWITCH CASE?
- Apakah kegunaan dari struktur perulangan? Jelaskan perbedaan struktur perulangan FOR dan WHILE!

```

#include<iostream>
using namespace std;

int main() {
    int B;
    cin >> B;

    if (B % 2 != 0) {
        cout << "satu: ";
        ++B;
        cout << B << endl;
    }
    if (B > 10) {
        cout << "dua: ";
        B--;
        cout << B << endl;
    } else {
        cout << "tiga: ";
        B = B + 2;
        if (B <= 7) {
            B = 8;
        } else if (B > 8) {
            B = B - 3;
        } else {
            B++;
        }
        cout << B << endl;
    }

    return 0;
}

```

# REVIEW PERCABANGAN

Apakah output program disamping, jika:

- a) B = 8
- b) B = 7
- c) B = 11
- d) B = 1

- a) Tiga : 7
- b) Satu : 8 , Tiga : 7
- c) Satu : 12, Dua : 11
- d) Satu : 2, Tiga : 8

# REVIEW PERULANGAN

```
#include<iostream>
using namespace std;

int main() {
    int B;
    cin >> B;

    while (B > 0) {
        B = B - 3;
        cout << B << endl;
    }

    return 0;
}
```

```
#include<iostream>
using namespace std;

int main() {
    int B;
    cin >> B;

    do {
        B = B - 3;
        cout << B << endl;
    } while (B > 0)

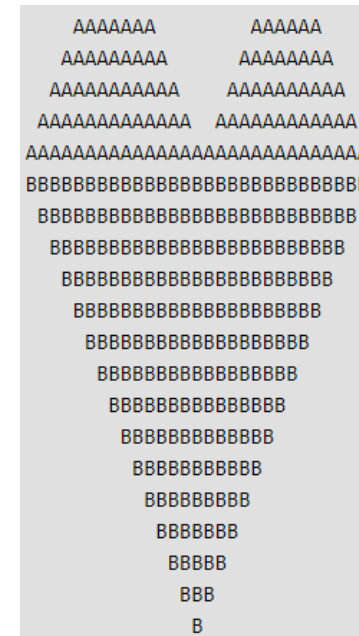
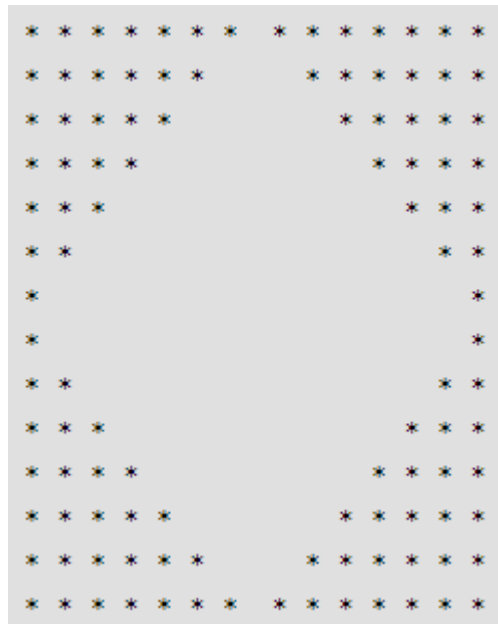
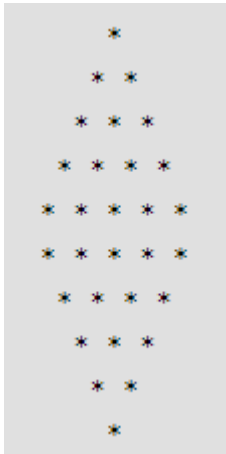
    return 0;
}
```

Apakah *output* program sama untuk semua masukan?

# PERULANGAN BERSARANG

Taukah kamu?

Dengan menggunakan bahasa pemrograman dan prinsip **perulangan bersarang**, kamu bisa lho membuat berbagai macam pola yang rapih seperti berikut ini:

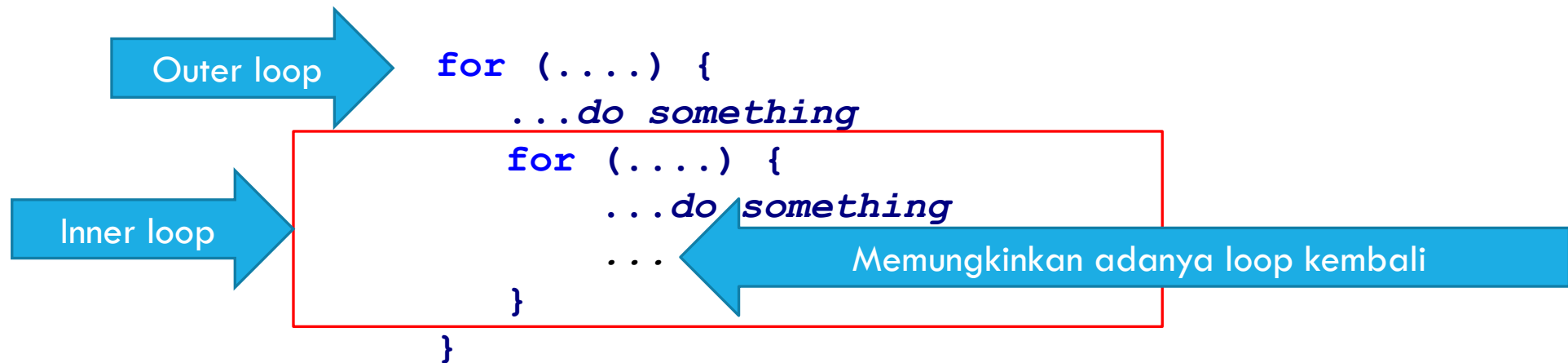


# PERULANGAN BERSARANG

Perulangan yang dilakukan di dalam kondisi perulangan terluar (outer loop)

Contoh penerapan: Pembangkit pola dan Algoritma pencarian data

Struktur umum



# CONTOH

Lakukan analisis terhadap algoritma program disamping!

Apakah yang tercetak di layar?

Coba bandingkan luaran yang tercetak di layar jika  $N = 3$  dan  $N = 5$ !

```
#include<iostream>
using namespace std;

int main() {
    int n;
    cout << "Masukan sebuah bilangan bulat: ";
    cin >> n;

    for (int i = n; i > 0; i--) {
        for (int j = i; j > 0; j--) {
            cout << "#";
        }
        cout << endl;
    }

    return 0;
}
```

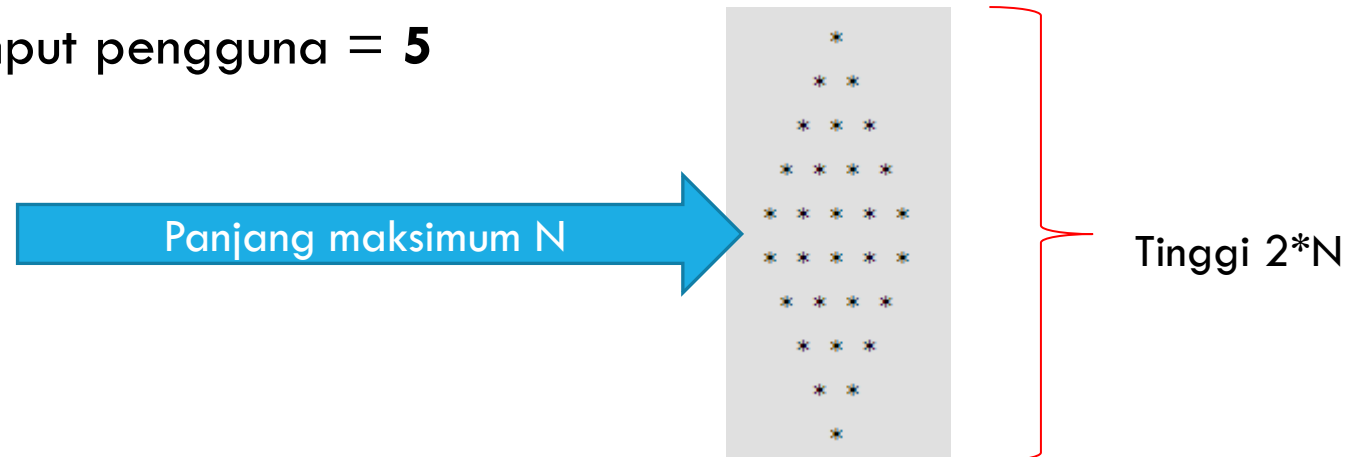
# LATIHAN

Deskripsi: Buatlah program yang dapat mencetak luaran berupa pola berlian seperti berikut ini

Input: sebuah bilangan bulat  $N$

Output: pola berlian dengan tinggi  $2*N$  dan panjang maksimum  $N$

Contoh luaran jika input pengguna = **5**





# PERCABANGAN DAN PERULANGAN

Jika anda diberi instruksi untuk membuat sebuah program yang menerima **masukan** sebuah bilangan bulat untuk kemudian **menentukan** dan **mencetak** ke layar apakah bilangan tersebut termasuk kategori bilangan genap atau ganjil, dan proses ini **diulang** terus menerus hingga pengguna memberi masukan bilangan -999 (program berhenti jika pengguna memasukan bilangan -999).

Struktur pemrograman seperti apa yang digunakan? Percabangan? Perulangan *while*, *for*, atau *do while*?

Bagaimanakah bentuk program tersebut? Coba anda buat program dan flowchart nya!

# CONTOH SOLUSI

```
#include<iostream>
using namespace std;

int main() {
    int n;
    cout << "Masukan sebuah bilangan bulat: ";
    cin >> n;

    while (n != -999) {
        if (n % 2 == 0) {
            cout << n << " termasuk bilangan GENAP" << endl;
        } else {
            cout << n << " termasuk bilangan GANJIL" << endl;
        }
        cout << "Masukan sebuah bilangan bulat: ";
        cin >> n;
    }

    return 0;
}
```

# LATIHAN

- Deskripsi: Buatlah sebuah program yang dapat melakukan pengecekan terhadap bilangan prima
  - Bilangan prima = Bilangan yang hanya memiliki faktor 1 dan bilangan itu sendiri.
  - Jika N habis dibagi oleh suatu bilangan M (yang bukan N), maka N **bukan** bilangan prima
- Input: sebuah bilangan bulat
- Output: tercetak ke layar kategori bilangan (prima atau bukan prima)

```

#include<iostream>
using namespace std;

int main() {
    int n;
    cout << "Masukan sebuah bilangan bulat: ";
    cin >> n;

    int i = 2;
    bool isPrime = true;

    while (i < n && isPrime)
    {
        if(n % i == 0 && n != 2 && n != 3)
        {
            isPrime = false;
        }
        i++;
    }
    if (isPrime) {
        cout << "Bilangan yang dimasukan adalah bilangan prima";
    } else {
        cout << "Bilangan yang dimasukan BUKAN bilangan prima";
    }
    return 0;
}

```

# CONTOH SOLUSI

**TERIMA KASIH**