



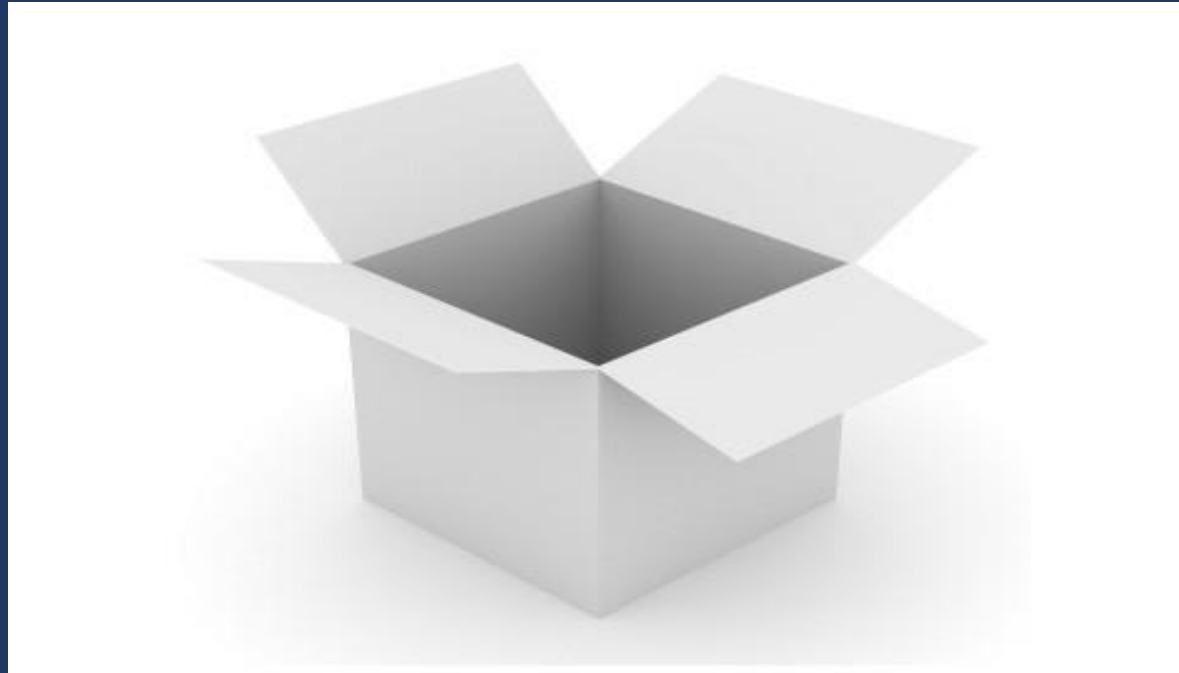
White Box Testing **#1**

Testing & Implementasi

Elkin Rilvani

Pertemuan 3

elkinrilvani@gmail.com



White Box Testing



Pengujian yang dilakukan dengan menggunakan kode program dari aplikasi yang ada ke dalam beberapa kasus pengujian

Some synonyms of white box testing are



Glass box Testing

**Clear box
Testing**

**Open box
Testing**

**Transparent
box Testing**

**Structural
Testing**

**Logic
driven
Testing**

**Design
based
Testing**



Metode White Box | Basis path testing

- Diusulkan oleh Tom McCabe pada tahun 1976
- Digunakan untuk mendapatkan ukuran kompleksitas logika
- Ukuran ini dijadikan sebagai panduan untuk menentukan jalur-jalur utama yang akan dieksekusi.





Memberikan jaminan bahwa semua jalur independen suatu modul digunakan minimal satu kali

Menggunakan semua keputusan logis untuk semua kondisi *true* atau *false*

Mengeksekusi semua perulangan pada batasan nilai dan operasional pada setiap kondisi.

Some types of white box testing are

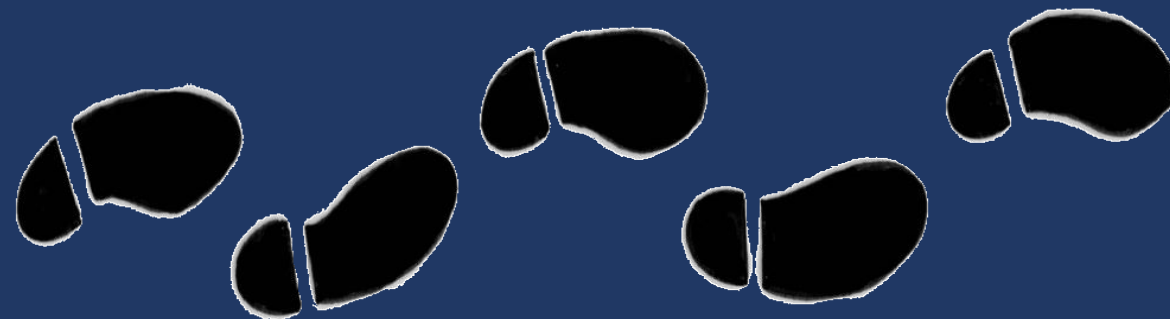
🔦 Basis Path Testing

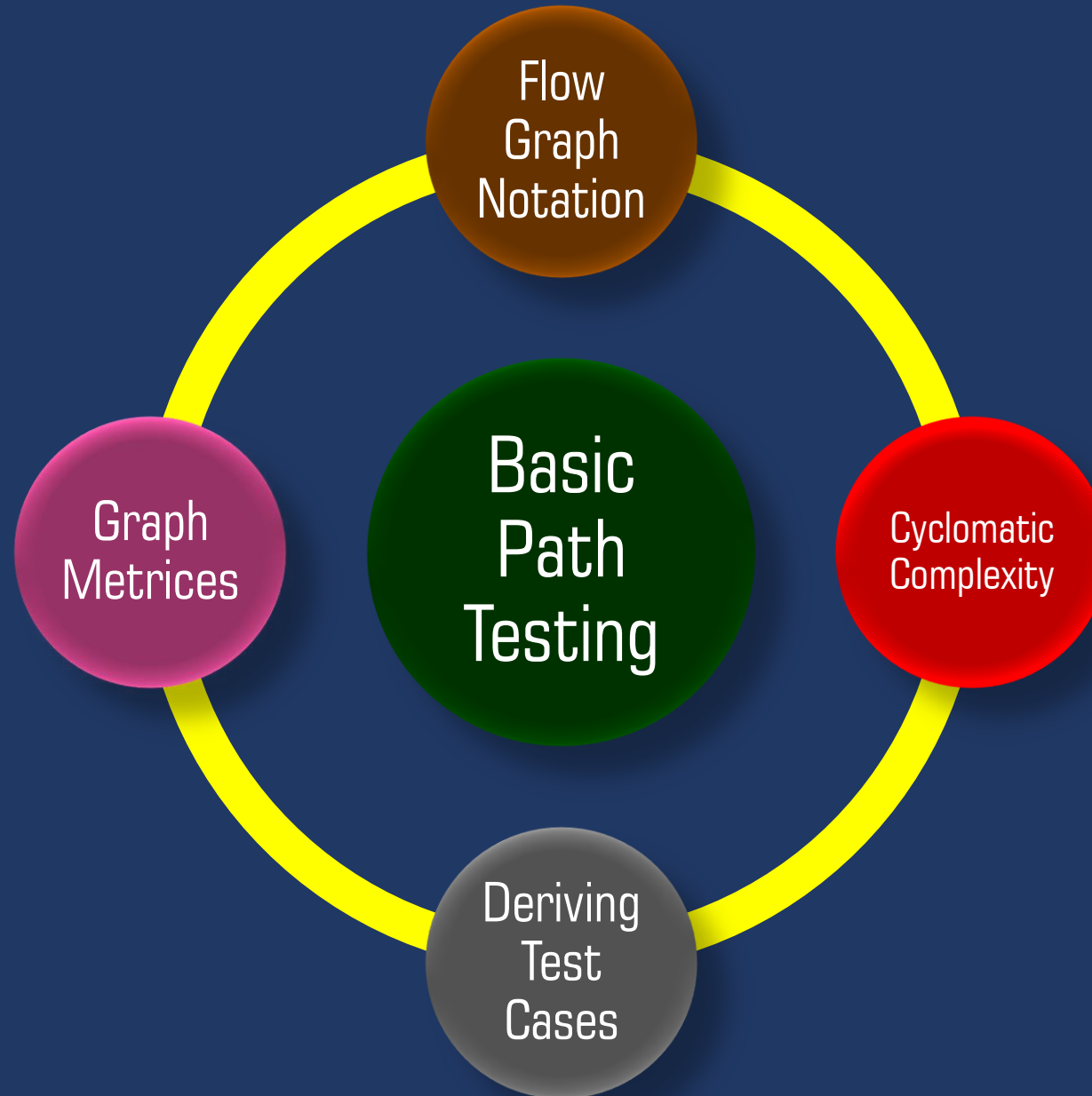
🔦 Condition Testing

🔦 Loop Testing

🔦 Control Structure

🔦 Data Flow Testing





Flow Graph Notation

Notasi sederhana yang digunakan untuk mewakili control flow disebut flow graph. Flow graph Mewakili arus kontrol logis yang digunakan untuk menggambarkan struktur pengendalian program.

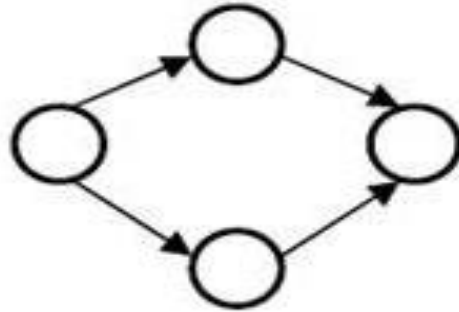
- ● Lingkaran (flow graph node)
- Menyatakan satu atau beberapa statement prosedural.
- → Panah (Edge / Link)
- Menyatakan aliran Kontrol / kendali.



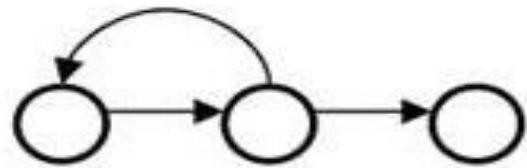
Standart Flow Graph In Coding



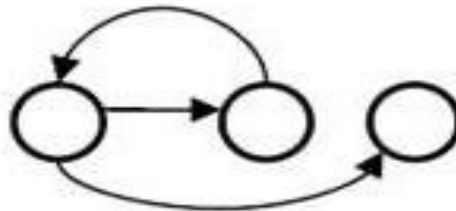
Sequence



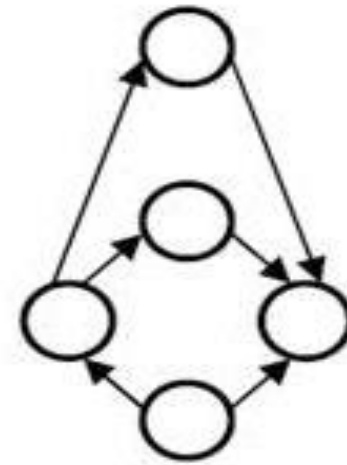
If



Until

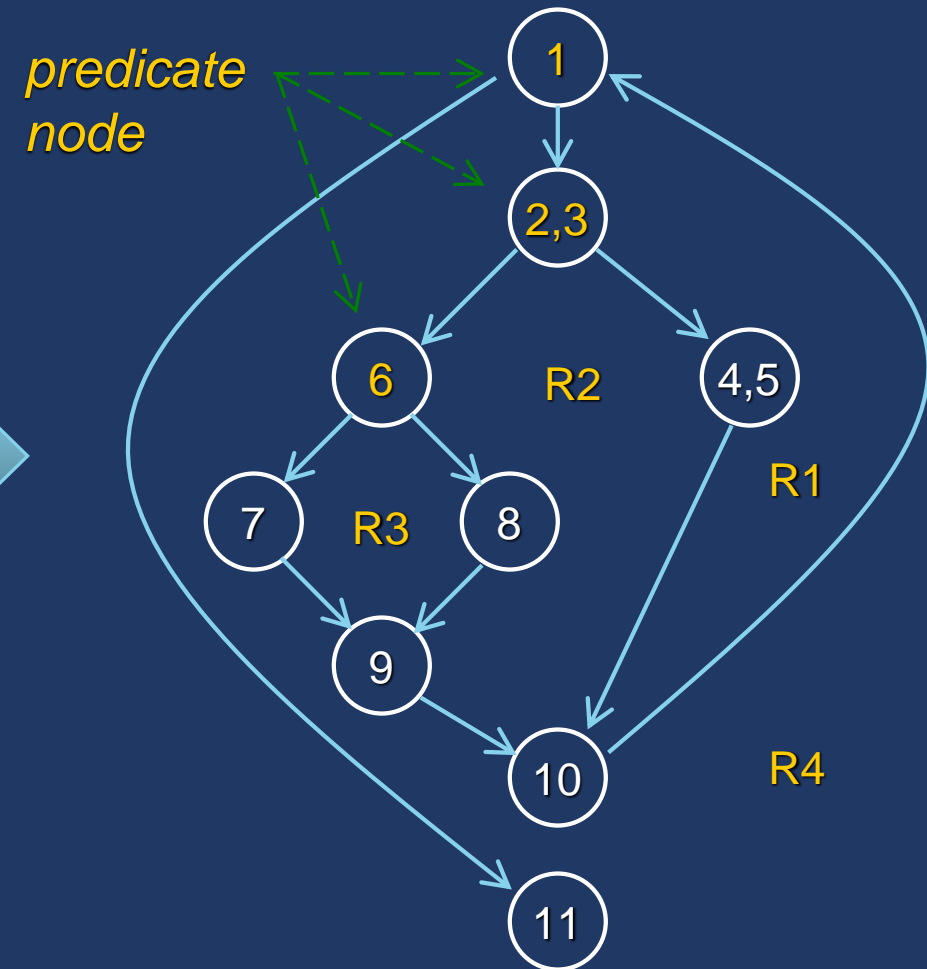
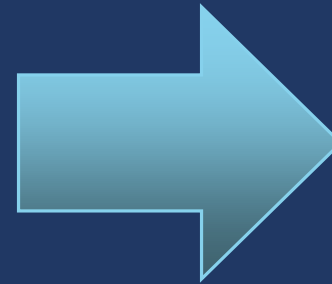
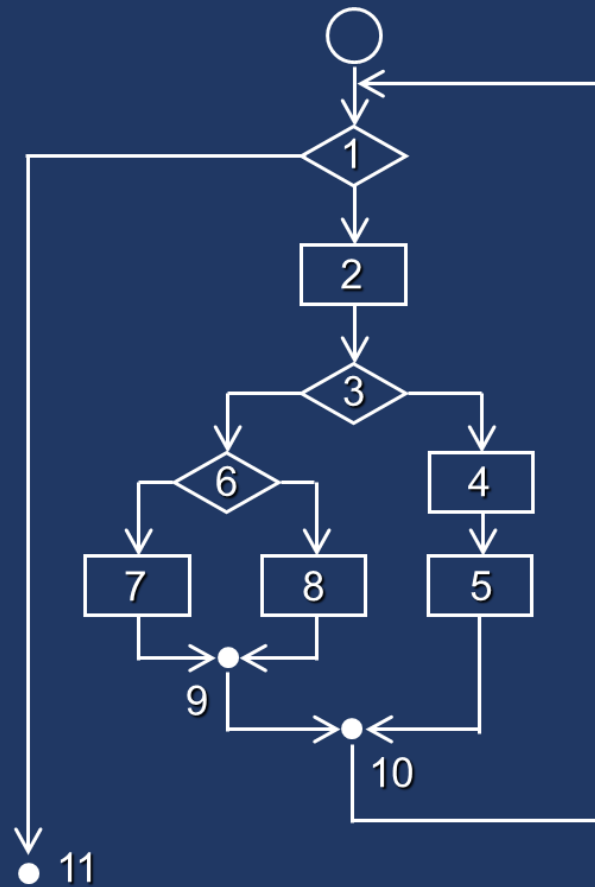


While



Case

Transformasi dari flow chart ke flow graph :





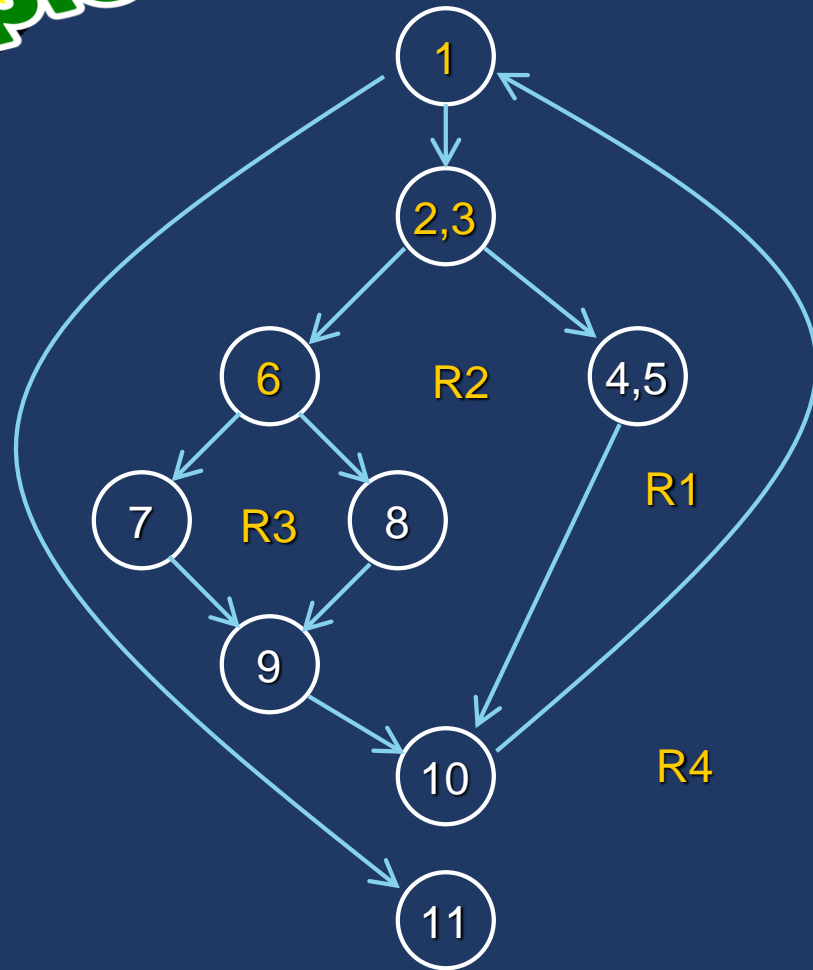
- Proses dan keputusan yg berurutan dimapping menjadi 1 node
- Setiap edge harus berakhir pada sebuah node (walaupun tdk merepresentasikan proses apapun)
- Region : daerah yang dibatasi oleh edge dan node
- Predicate node : node yg merupakan kondisi keluar (2 atau lebih edge akan keluar dari sini)

Cyclomatic Complexity

Kompleksitas Siklomatis adalah metrics perangkat lunak yang memberikan pengukuran kuantitatif terhadap kompleksitas logis suatu program, nilai yang didapat akan menentukan jumlah jalur independen dalam himpunan path,

Jalur independen adalah jalur yang terdapat dalam program yang mengintroduksi sedikitnya satu rangkaian pernyataan proses atau kondisi baru.





■ Jalur independen :

- jalur 1 : 1-11
- jalur 2 : 1-2-3-4-5-10-1-11
- jalur 3 : 1-2-3-6-8-9-10-1-11
- jalur 4 : 1-2-3-6-7-9-10-1-11

■ Bukan Jalur independen :

- 1-2-3-4-5-10-1-2-3-6-8-9-10-1-11
karena merupakan kombinasi dari paths yang sudah didefinisikan

Calculate

- Perhitungan matematis *cyclomatic complexity* – CC atau $V(G)$, for flow graph :

- $V(G) = \text{jumlah region}$
- $V(G) = E - N + 2$
- $V(G) = P + 1$, dimana P – *predicate node*

- Contoh perhitungan :

- $V(G) = 4$, ada 4 *region* R1, R2, R3, R4
- $V(G) = 11 \text{ edges} - 9 \text{ nodes} + 2 = 4$
- $V(G) = 3 \text{ predicate node} + 1 = 4$



Deriving Test Cases

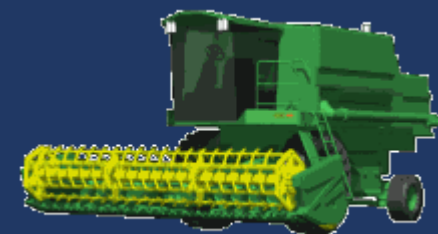
Tujuan utama basis path testing adalah untuk mendapatkan kasus uji untuk prosedur yang diuji. Pengujian basic path dapat dilihat sebagai serangkaian langkah

Step 1: Buat flow graph yang sesuai dari desain atau kode sumber yang diberikan.

Step 2: Hitunglah kompleksitas cyclomatic dari flow graph ini dengan menggunakan salah satu dari ketiga formula tersebut.

Step 3: Tentukan satu set dasar jalur independen linier.

Step 4: Siapkan uji kasus yang akan memaksa eksekusi setiap jalur di basis set.

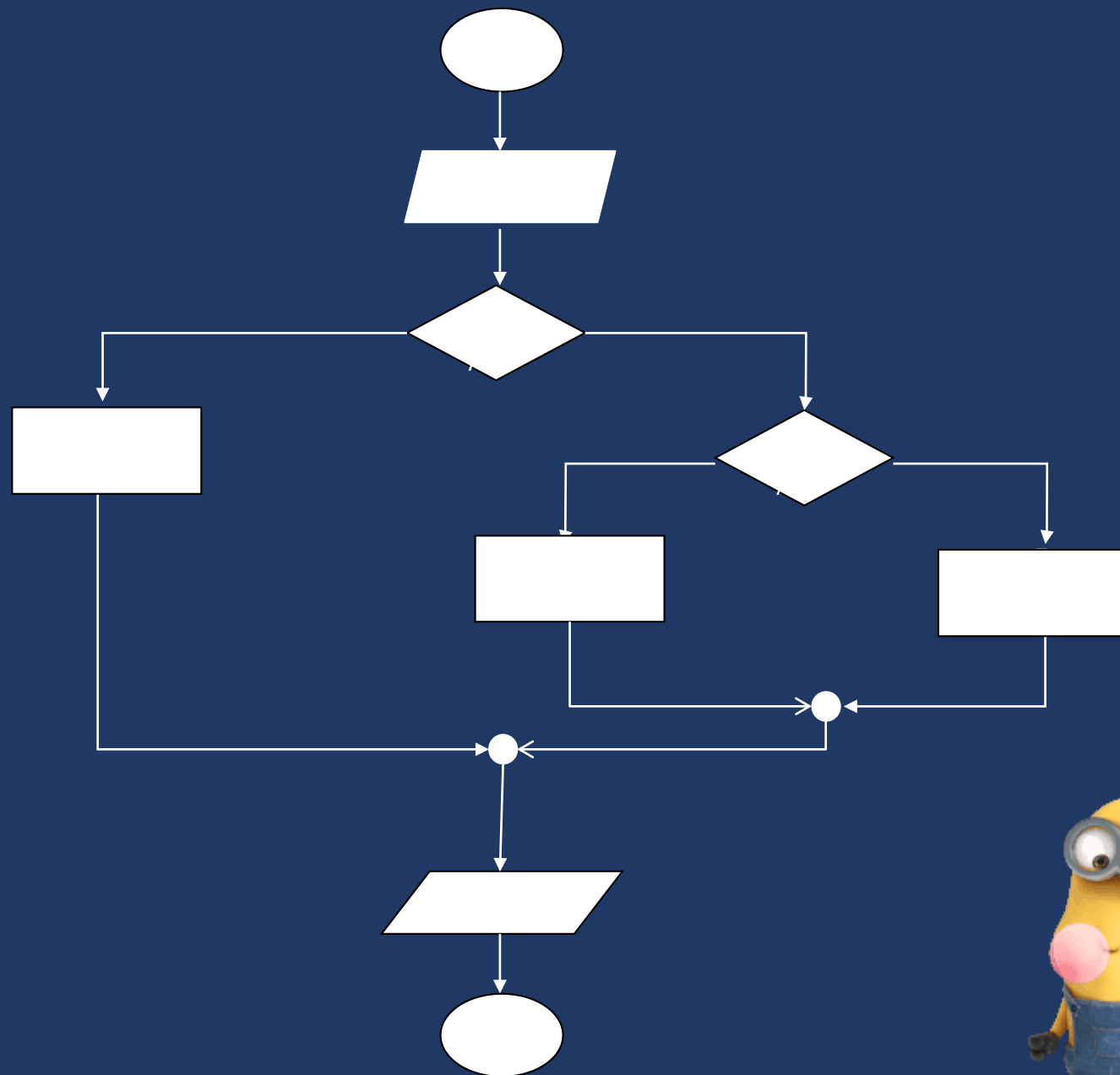




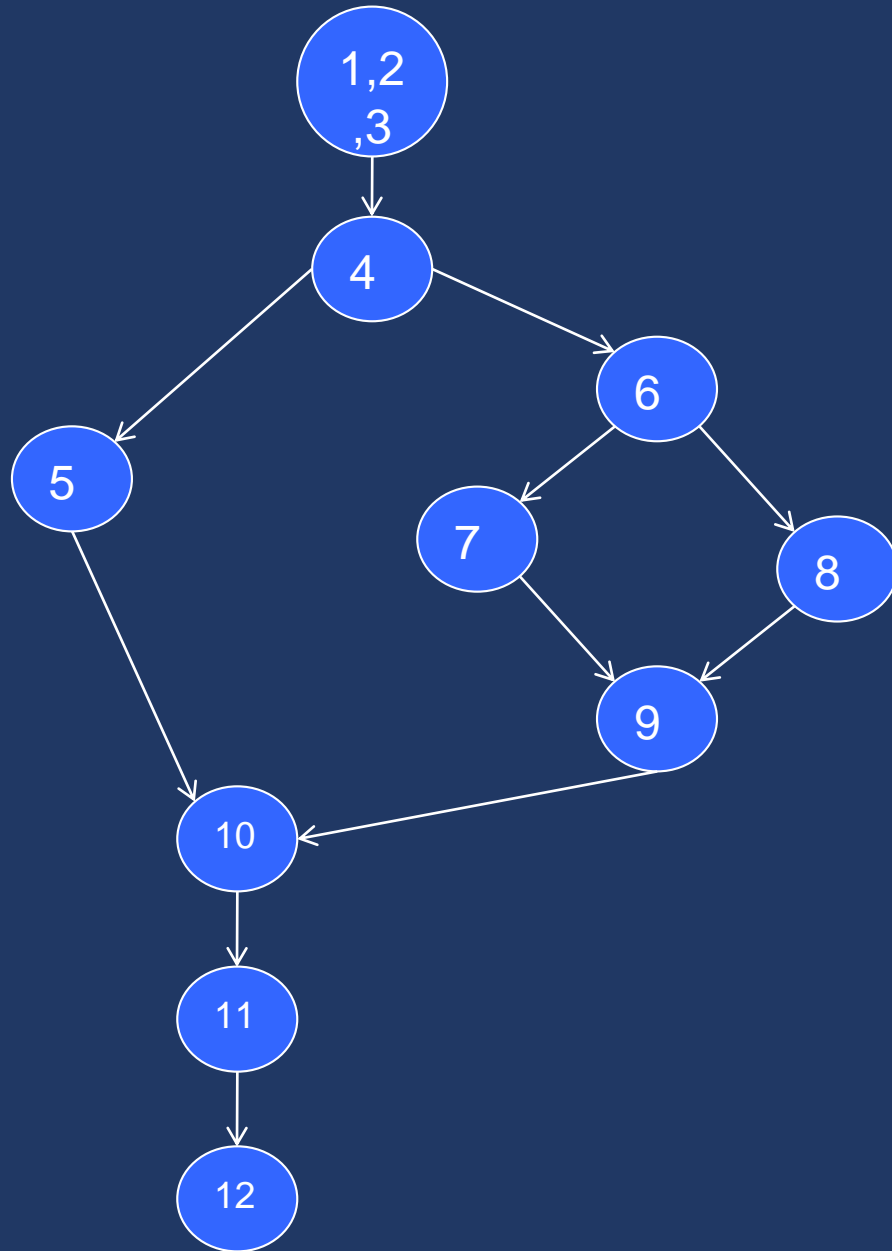
```
Var
A, B, C : integer
Begin
A := 10;           (1)
B := 5;            (2)
C := 6;            (3)
    If A > B        (4)
        then C := A + B (5)
        Else if A < C (6)
            then C := A (7)
            Else C := B; (8)
        Endif       (9)
    Endif           (10)
    Println('Nilai C = ', C); (11)
End                (12)
```

THE
NEXT
STEP

■ Convert to Flowchart



NEXT STEP



■ Flow Graph



Jalur 1 : 1,2,3 – 4 – 5 – 10 – 11 – 12

Jalur 2 : 1,2,3 – 4 – 6 – 7 – 9 – 10 – 11 – 12

Jalur 3 : 1,2,3 – 4 – 8 – 9 – 10 – 11 – 12

Procedure Deriving Test Cases:

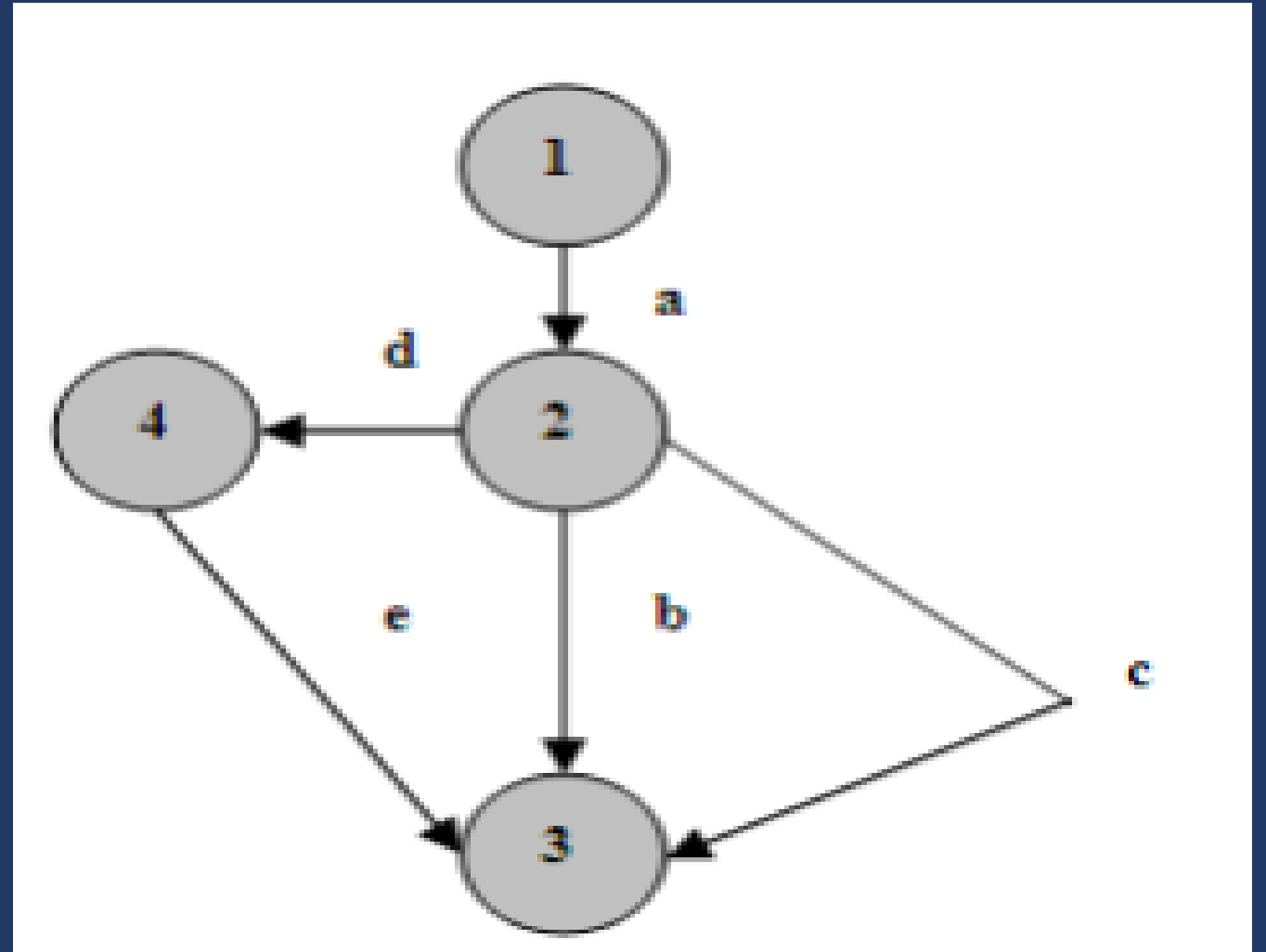
Jalur 1: `value(C) = valid, jika $A > B$ untuk menghasilkan $C=A+B$`

Jalur 2: `value(C) = tidak valid, jika $A > B$ untuk menghasilkan $C=A$
 value(C) = valid, jika $A < C$ untuk menghasilkan $C=A$`

Jalur 3: `value(C) = tidak valid, jika $A > B$
 value(C)= tidak valid, jika $A > C$ untuk menghasilkan $C=B$
 value(C)= valid, jika A tidak besar dari B dan juga tidak
 besar dari C untuk menghasilkan $C=B$`

Graph Metrics

Graph matrix adalah matriks dua dimensi yang membantu dalam menentukan himpunan dasar. Ini memiliki kolom dan baris masing-masing sama dengan jumlah simpul dalam flow graph. Untuk membedakan satu sama lain setiap simpul diwakili oleh beberapa huruf. Setiap tepi dilengkapi dengan beberapa bobot link (0-tanpa koneksi, 1-jika ada koneksi).





GetResponse



Table 1: Graph Matrix

| Node | 1 | 2 | 3 | 4 |
|------|---|---|------|---|
| 1 | | a | | |
| 2 | | | b, c | d |
| 3 | | | | |
| 4 | | | e | |

Matriks grafis selanjutnya disebut sebagai matriks koneksi, dan digambarkan serupa dengan Graph Matrix dengan memperhatikan arah in-out dari edge.



Table 2: Connection Matrix

| Node | 1 | 2 | 3 | 4 | Connections |
|------|---|---|------|---|-------------|
| 1 | | 1 | | | $1-1=0$ |
| 2 | | | 1, 1 | 1 | $3-1=2$ |
| 3 | | | | | 0 |
| 4 | | | 1 | | $1-1=0$ |

Cyclomatic Complexity is $2 + 1 = 3$

Thank
you