

REKAYASA PERANGKAT LUNAK <SOFTWARE TESTING>

Antonius Bima Murti Wijaya S.T.,M.T

TODAY'S TOPIC

Software Quality Assurance(SQA)

Software Testing Strategy

White Box Testing

Black Box Testing

Testing for specialized environment, Architectures and application

QUALITY? WHAT IS QUALITY?

- Ketika produk memenuhi kebutuhan
- Berapa banyak pelanggan yang bersedia membayar untuk suatu produk

**BETTER TO IMPROVE CUSTOMER WILLING TO PAY RATHER
DELIVER CHEAP PRODUCT WITH LESS QUALITY**

SOFTWARE QUALITY

- Sebuah **proses software yang efektif**, diterapkan dengan cara membuat sebuah **software yang berguna** dan **menghasilkan nilai yang terukur** untuk pembuat dan pemakai.
 - Proses software yang efektif melibatkan proses cek and balances, analisa yang mudah terhadap masalah, penanganan perubahan, technical reviews.
 - Software yang berguna berarti memberikan kebutuhan user, memberikan software yang tangguh atau error-free.
 - Menghasilkan nilai yang terukur berarti sama-sama memberi keuntungan baik bagi pengembang(maintenance, bugs, support) dan user.

ELEMENT OF SQA

- Standards
 - SQA memastikan bahwa standar yang ada seperti IEEE dan ISO sudah diadaptasi dan nyaman
- Reviews and audits
 - Memastikan guideline dan SOP sudah dijalankan dengan benar
- Testing
 - Memastikan bahwa pengujian sudah direncanakan dengan benar
- Error collection and analysis
 - SQA mengumpulkan dan menganalisis bagaimana error dapat terjadi dan bagaimana mengurangnya
- Change management
 - Change can lead to confusion and poor quality

ELEMENT OF SQA

- Education
 - Develop and conduct training and education for software developer
- Vendor management
 - Memastikan software berkualitas baik diperoleh dari standar kualitas yang harus di penuhi oleh vendor
- Security management
 - Memastikan sudah mengikuti standar keamanan untuk mengamankan data
- Safety
 - Mengukur dampak dari kegagalan software untuk mengurangi resiko
- Risk management
 - Memastikan manajemen resiko dan perencanaanya sudah dibangun

SOFTWARE TESTING STRATEGY

- Sebuah roadmap yang mendeskripsikan langkah langkah yang harus dibangun sebagai bagian dari pengujian
- Pengujian merupakan sekumpulan aktivitas yang dapat **direncanakan dan dibangun** secara sistematis
- Testing terdiri dari proses verifikasi dan validasi
 - Verifikasi “are we building the product right? ”
 - Validation “are we building the right product?”

STRATEGIC ISSUE

- Tentukan kebutuhan produk didalam cara yang terhitung jauh sebelum pengujian terjadi
- Obyektif pengujian harus jelas(dapat diukur)
- Memahami pengguna software dan membentuk profil dari setiap kategori pengguna
- Mengembangkan rencana pengujian yang menegaskan “rapid cycle testing”-> fast response and feedback
- Membangun software yang tangguh yang didesain untuk menguji dirinya sendiri
- Menggunakan review teknik sebagai penyaring utama untuk testing

SYSTEM TESTING THAT ARE WORTHWHILE FOR SOFTWARE BASED SYSTEM

- Recovery testing
 - Uji sistem yang memaksa perangkat lunak gagal dalam berbagai cara dan memverifikasi bahwa pemulihan dilakukan dengan benar
- Security Testing
 - Selama pengujian keamanan, tester memainkan peran (s) dari individu yang ingin menembus sistem
- Stress Testing
 - Hadapi program dengan situasi abnormal seperti kuantitas, frekuensi, atau volume
- Performance testing
 - Uji semua fungsi untuk memenuhi output keinginan, dengan mendokumentasikan sistem, penguji dapat mengungkap situasi yang menyebabkan degradasi dan kemungkinan kegagalan sistem

- Deployment testing
 - Latih perangkat lunak di setiap lingkungan untuk mengoperasikan ex: uji sistem di berbagai OS, Web Browser, platforms

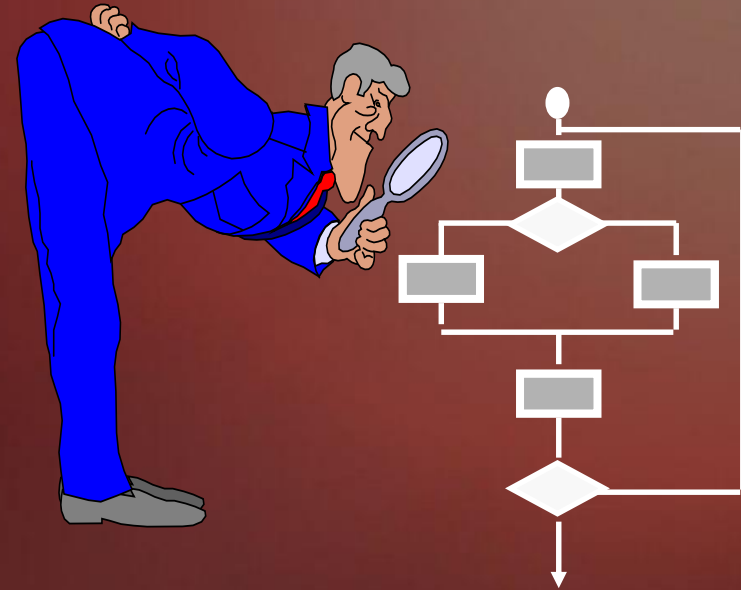
CHARACTERISTIC OF TESTABILITY “HOW EASILY SOFTWARE CAN BE TESTED”

- Operabilitas "semakin baik kerjanya semakin efisien maka dapat diuji"
- Pengamatan "semua yang bisa Anda lihat adalah semua yang bisa Anda uji"
- Controllablity “Semakin baik kita dapat mengontrol perangkat lunak, semakin banyak pengujian dapat diotomatisasi dan dioptimalkan”
- Dekomposisi "mengontrol ruang lingkup, kita dapat lebih cepat mengisolasi masalah dan melakukan pengujian ulang yang lebih cerdas"
- Kesederhanaan "semakin sedikit ada tes, semakin cepat kita bisa mengujinya"
- Stabilitas “semakin sedikit perubahan, semakin sedikit gangguan untuk menguji”
- Keterbukaan “semakin banyak informasi yang kami miliki, semakin pintar kami akan menguji”

WHITE BOX TESTING

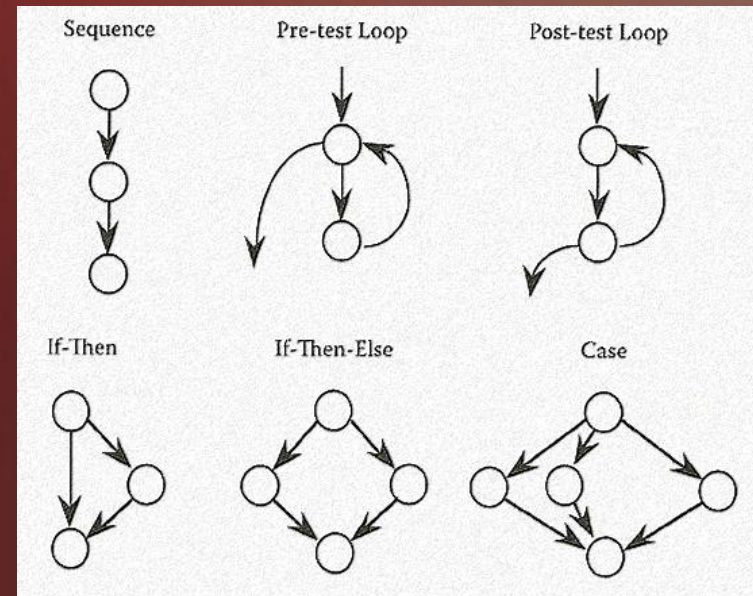
- Menguji dengan memperhatikan mekanisme internal sistem
- Menguji untuk memastikan operasi internal berjalan sesuai spesifikasi
- Semua komponen diuji

... our goal is to ensure that all statements and conditions have been executed at least once ...



TEKNIK-TEKNIK WHITE-BOX TESTING

- Basis path testing
 - Digunakan untuk memperoleh pengukuran kerumitan logic dari prosedur perancangan dan menggunakan perhitungan ini sebagai langkah untuk mendefinisikan kumpulan langkah eksekusi



THEN HOW TO DERIVE THE WHITE BOX TEST

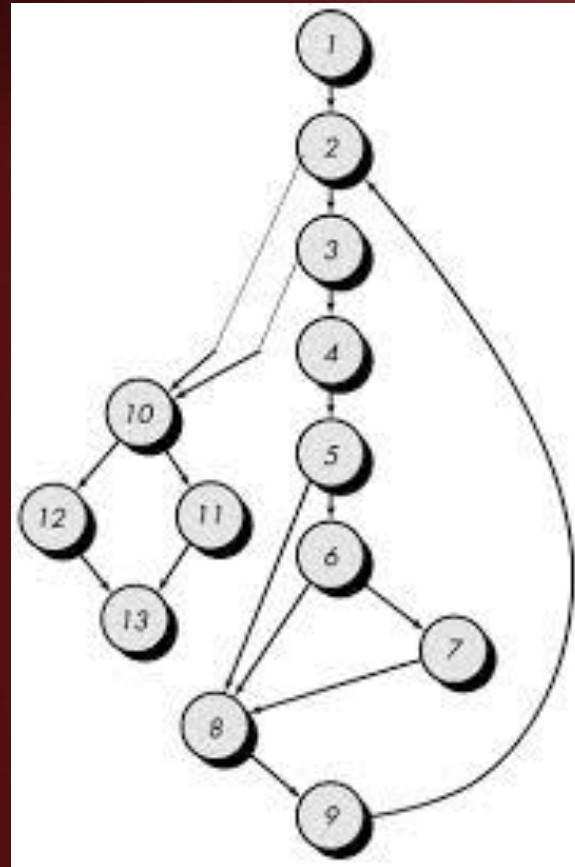
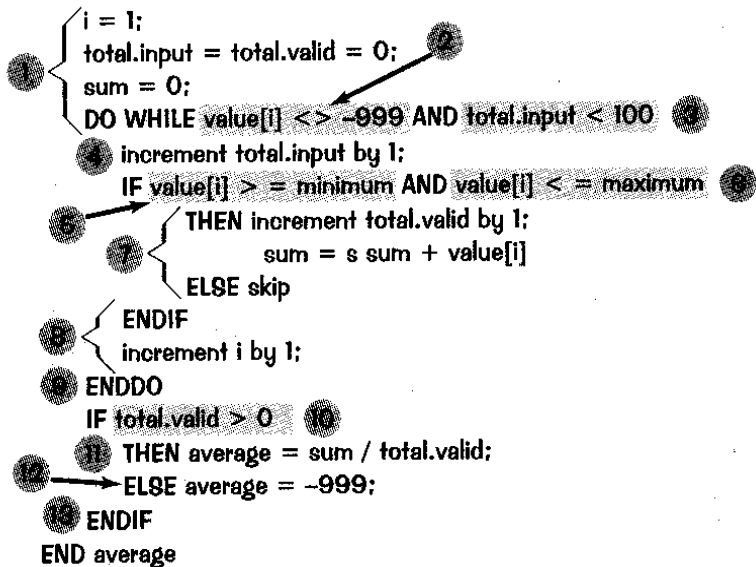
- Use the design or code as a foundation
- Determine a basis set of path

PROCEDURE average:

- * This procedure computes the average of 100 or fewer numbers that lie between bounding values; it also computes the sum and the total number valid.

INTERFACE RETURNS average, total.input, total.valid;
INTERFACE ACCEPTS value, minimum, maximum;

TYPE value[1:100] IS SCALAR ARRAY;
TYPE average, total.input, total.valid;
minimum, maximum, sum IS SCALAR;
TYPE i IS INTEGER;



DRAW THE FLOW GRAPH FOR TEST CASE

```
Int c=2;
For (int i=0;i<5;i++)
{
    if (c==2)
        a=a+c;
    else
        c=c+a;
}
If (a>10)
    System.out.println("hello");
Else
    System.out.println("bye");
```

OTHER EXAMPLE

```
public void f() {  
    if(count > 10){  
        fixed_count = fixed_count + count;  
        done = 1;  
    } else if(count >5){  
        fixed_count --;  
    } else {  
        fixed_count = count * 4;  
    }  
}
```