

LAPORAN

PRAKTIKUM SOFTWARE QUALITY ASSURANCE

Pertemuan ke – 2



Disusun Oleh :

Rafiqul Huda

185610068

LABORATORIUM TERPADU
SEKOLAH TINGGI MANAJEMEN INFORMATIKA & KOMPUTER
AKAKOM YOGYAKARTA

2020

BAB I

PRAKTIK

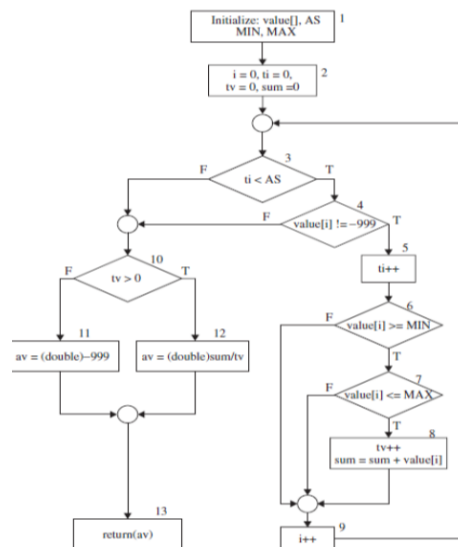
1. Kode Program :

```

1 package javaapplication1;
2 import static javaapplication1.binary.binsearch;
3 public class lat {
4     public static double ReturnAverage(int value[], int AS, int MIN, int MAX){
5         int i, ti, tv, sum;
6         double av;
7         i = 0; ti = 0; tv = 0; sum = 0;
8         while (ti < AS && value[i] != -999) {
9             ti++;
10            if (value[i] >= MIN && value[i] <= MAX) {
11                tv++;
12                sum = sum + value[i];
13            }
14            i++;
15        }
16        if (tv > 0)
17            av = (double)sum/tv;
18        else
19            av = (double) -999;
20        return (av);
21    }
22    public static void main(String[] args) {
23        int[] value= {60,80,70,75,65,75,85,80,90,70};
24        System.out.println(ReturnAverage(value,10,60,70));
25    }
26 }

```

2. CFG



3. Path independent

Path 1 1-2-3(F)-10(T)-12-13

Path 2 1-2-3(F)-10(F)-11-13

Path 3 1-2-3(T)-4(T)-5-6(T)-7(T)-8-9-3(F)-10(T)-12-13

Path 4 1-2-3(T)-4(T)-5-6(T)-7(T)-8-9-3(T)-4(T)-5-6(T)-7(T)-8-9-3(F)-10(T)-12-13

4. Hasil Uji

```
run:
66.25
BUILD SUCCESSFUL (total time: 0 seconds)
```

5. Penjelasan

Merupakan program untuk menghitung rata-rata dimana inputan yang dimasukkan adalah value[] (nilai-nilai array), AS (jumlah Array), MIN (nilai terendah dalam array), dan MAX (nilai maksimal dalam array).

Ada program returnaverage ini dimasukkan data berupa nilai array yaitu (60,80,70,75,65,75,85,80,90,70), yang mana memiliki jumlah 10, mempunyai nilai MIN 60, dan nilai MAX 90. Sehingga menghasilkan nilai seperti pada nomor 4.

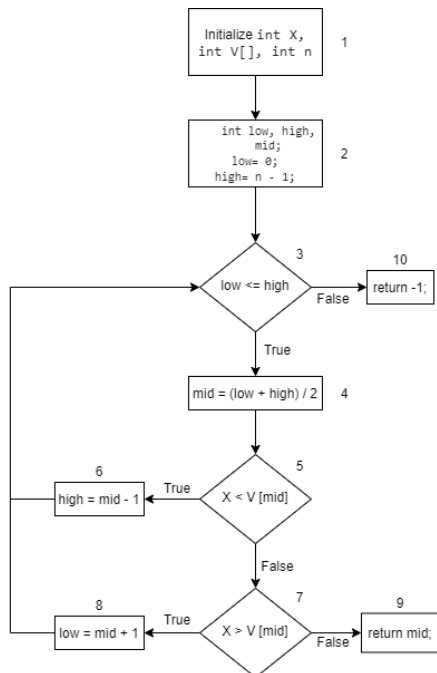
BAB II

LATIHAN

Kode Program Binary Search :

```
1 package javaapplication1;
2 public class binary {
3     static int binsearch(int X, int V[], int n)
4     {
5         int low, high, mid;
6         low = 0;
7         high = n - 1;
8         while (low <= high) {
9             mid = (low + high)/2;
10            if (X < V[mid])
11                high = mid - 1;
12            else if (X > V[mid])
13                low = mid + 1;
14            else
15                return mid;
16        }
17        return -1;
18    }
19    public static void main(String[] args) {
20        int[] array = {1,2,3,4,5,6,7,8,9};
21        System.out.println(binsearch(5, array, 9));
22    }
23 }
```

1. CFG Program Binary Search :

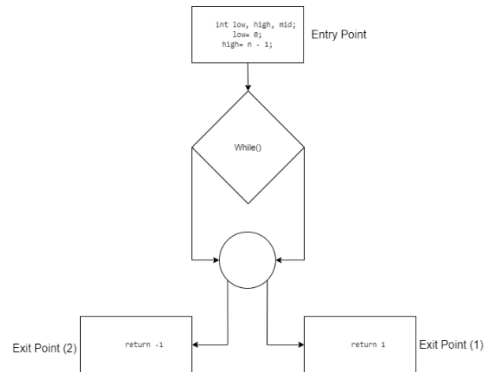


1. Pendeklarasian variabel x, V[], dan n
2. mendeklarasikan dan memberikan nilai pada variabel low, high, dan mid.
3. kondisi dimana low <= high, bila true lanjut ke nomor 4, false lanjut ke nomor 10
4. kondisi dimana mid = (low+high)/2 bila true lanjut nomor 5

5. kondisi dimana $X < V[\text{mid}]$ apabila true lanjut nomor 6, false lanjut nomor 7

7. kondisi dimana $X > [\text{mid}]$ apabila true lanjut nomor 8, false nomor 9

2. Jalur Entry dan Exit



3. Jalur Independent

Path 1 -> 1 -2- 3(T)-5(T)-6

Path 2 -> 1 -2- 3(T)-5(F)-7(T)-8

Path 3 -> 1 -2- 3(T)-5(F)-7(F)-9

Path 4 -> 1 -2- 3(F)-10

4. Hasil Uji

```
run:
4
BUILD SUCCESSFUL (total time: 0 seconds)
```

Hasil Ketika diinputkan $x = 5$ dan $n = 9$, menghasilkan jark berupa nilai angka 4.

5. Penjelasan

Merupakan program untuk mencari indeks pada array dengan menggunakan binary search. Dimana program ini menggunakan perulangan while untuk mencari nilai tengah(mid), tertinggi(high), dan terendah(low). Sehingga program ini memiliki 4 kondisi percabangan yang mana sesuai dengan path independent pada nomor 2.

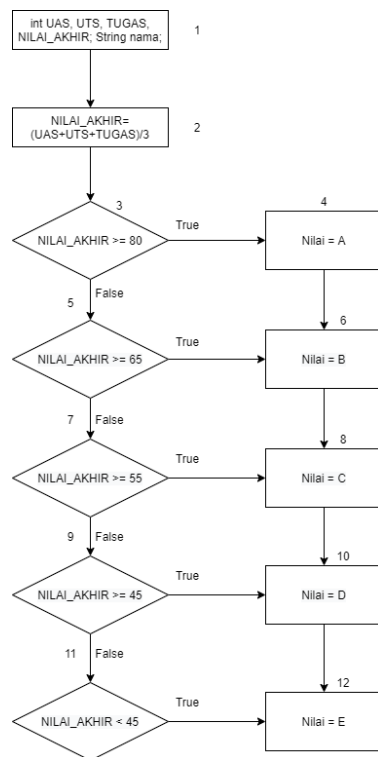
BAB III

TUGAS

Kode Program :

```
1 package javaapplication1;
2 import java.util.Scanner;
3 public class Latihan4 {
4     public static void main(String[] args) {
5         int UAS, UTS, TUGAS, NILAI_AKHIR;
6         String nama;
7         Scanner keyboard= new Scanner(System.in);
8         System.out.println("Masukkan Nama");
9         nama=keyboard.nextLine();
10        System.out.println("Masukkan Nilai Tugas");
11        TUGAS=keyboard.nextInt();
12        System.out.println("Masukkan Nilai UTS");
13        UTS=keyboard.nextInt();
14        System.out.println("Masukkan Nilai UAS");
15        UAS=keyboard.nextInt();
16        NILAI_AKHIR= (UAS+UTS+TUGAS)/3;
17
18        if(NILAI_AKHIR >= 80){
19            System.out.println("Nilai Akhir = A");
20        }else if(NILAI_AKHIR >= 65){
21            System.out.println("Nilai Akhir = B");
22        }else if(NILAI_AKHIR >= 55){
23            System.out.println("Nilai Akhir = C");
24        }else if(NILAI_AKHIR >= 45){
25            System.out.println("Nilai Akhir = D");
26        }else{
27            System.out.println("Nilai Akhir = E");
28        }
29    }
30 }
```

1. CFG



1. pendeklarasian variabel UAS, UTS, TUGAS, dan NILAI_AKHIR

2. memberikan nilai pada variabel NILAI_AKHIR dimana nilainya adalah (UAS+UTS+TUGAS)/3.
3. kondisi if apabila NILAI_AKHIR >= 80 bila true lanjut ke nomor 4, false lanjut ke nomor 5
5. kondisi if apabila NILAI_AKHIR >=65 bila true lanjut ke nomor 6, false lanjut ke nomor 7
7. kondisi if apabila NILAI_AKHIR >=55 bila true lanjut ke nomor 8, false lanjut ke nomor 9
9. kondisi if apabila NILAI_AKHIR >=45 bila true lanjut ke nomor 10, false lanjut ke nomor 11
11. kondisi if apabila NILAI_AKHIR <45 bila true lanjut ke nomor 12

2. Jalur Independen

Path 1 -> 1-2-3(T)-4

Path 2 -> 1-2-3(F)-5(T)-6

Path 3-> 1-2-3(F)-5(F)-7(T)-8

Path 4-> 1-2-3(F)-5(F)-7(F)-9(T)-10

Path 5-> 1-2-3(F)-5(F)-7(F)-9(F)-11(T)-12

3. Hasil Uji

- Kondisi 1 dimana nilai >=80 menapat nilai A

```
run:
Masukkan Nama =
rafiqul
Masukkan Nilai Tugas =
80
Masukkan Nilai UTS =
80
Masukkan Nilai UAS =
80
Nama Mahasiswa = rafiqul
Nilai AKhir = A
BUILD SUCCESSFUL (total time: 11 seconds)
```

- Kondisi 2 dimana nilai >=65 mendapat nilai B

```
run:
Masukkan Nama =
rafiqul
Masukkan Nilai Tugas =
65
Masukkan Nilai UTS =
65
Masukkan Nilai UAS =
65
Nama Mahasiswa = rafiqul
Nilai Akhir = B
BUILD SUCCESSFUL (total time: 15 seconds)
```

- Kondisi 3 dimana nilai ≥ 55 mendapat nilai C

```
run:
Masukkan Nama =
rafiqul
Masukkan Nilai Tugas =
55
Masukkan Nilai UTS =
55
Masukkan Nilai UAS =
55
Nama Mahasiswa = rafiqul
Nilai Akhir = C
BUILD SUCCESSFUL (total time: 8 seconds)
```

- Kondisi 4 dimana nilai ≥ 45 mendapat nilai D

```
run:
Masukkan Nama =
rafiqul
Masukkan Nilai Tugas =
45
Masukkan Nilai UTS =
45
Masukkan Nilai UAS =
45
Nama Mahasiswa = rafiqul
Nilai Akhir = D
BUILD SUCCESSFUL (total time: 6 seconds)
|
```

- Kondisi 5 dimana nilai < 45 mendapat nilai E

```
run:
Masukkan Nama =
rafiqul
Masukkan Nilai Tugas =
30
Masukkan Nilai UTS =
20
Masukkan Nilai UAS =
10
Nama Mahasiswa = rafiqul
Nilai Akhir = E
BUILD SUCCESSFUL (total time: 8 seconds)
|
```

4. Penjelasan

Merupakan program untuk menghitung nilai akhir dari mahasiswa. Dimana nilai akhir tersebut diambil dari rata-rata nilai tugas, UTS, dan UAS. Nilai mahasiswa terdiri dari rentang nilai A sampai E. dimana kondisi A adalah bahwa nilai akhir ≥ 80 , B nilai ≥ 65 , C nilai ≥ 55 , nilai D ≥ 45 , dan E apabila mendapat nilai ≤ 40 .

Untuk membuat program tersebut kita menggunakan kondisi if else dimana akan ada 5 kondisi. Begitu juga dengan path yang dihasilkan akan sesuai dengan kondisi tersebut yaitu 5. Kemudian setelah semua kondisi diuji coba (pada no 4) hasilnya sesuai dengan setiap kondisi.

BAB IV

KESIMPULAN

Dengan melakukan praktikum kali ini mahasiswa dapat memahami mekanisme pengujian aliran kendali dan membuatnya pada program lain. Mahasiswa bahwa dengan menggunakan pengujian aliran program maka program akan terjamin kualitasnya. Ketika melakukan pengujian maka programmer akan tahu apabila terdapat error pada programnya, dengan cara melewati setiap kondisi-kondisi pada program.