Izzatiqisyah.

declare → int [] n;
initial + create → n = new int [10];

int [] n = new int [10];
= {1, 2, 3};
bila dah initial, tak perlu create. program dah fahom size tu 3.

Tutorial 5

Array and Array Processing I

Section A: Self-Test

1. What is an array?

An array is used to store a collection of data within the same data type. ✓

2. Write a Java statement to declare an array that can store 10 integers.

int [] numbers = new int [10] ✓

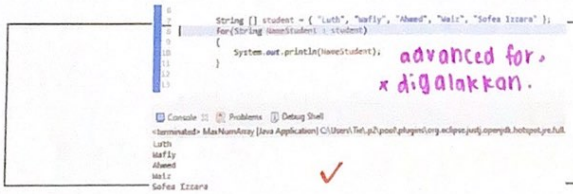
declare : int [] numbers;
declare + create : int [] numbers = new int [10];

3. Declare and initialize an array of strings named students to store the following names: Luth, Wafiy, Ahmed, Waiz, and Sofea Izzara. declare and initialize.

(String [] student = { "Luth", "Wafiy", "Ahmed", "Waiz", "Sofea Izzara" };) ✓

key a array.

4. Print the students array (refer Q3)



advanced for. x digalakkan. ✓

nama array size of array

```
for (int i = 0; i<student.length ; i++)
{
System.out.print(student[i] + " ");
}
```

Section B: Hand Tracing

1. Given the following declaration that represents maximum daily temperature in Bandar Baru Bangi from 21st to 30th October.

1 2 3 4 5 6

```
int [] temp = {33, 30, 32, 32, 33, 33, 33, 30, 30, 32};
```

Answer the following questions:

nama array = temp
datatype = int
size = 10
max index = 9

a. What does `temp[0]` , `temp[3]` and `temp[8]` represent?

temp[0] = 33
temp[3] = 32
temp[8] = 30 ✓

array.

inoutput → for
process → for
output → for

```
for (int i=0 ; i = student.length ; i++)
{
S.O.P(student[i] + "  ");
}
```

b. What is the temperature on Wednesday, 26th Oct?

(33) ✓

c. What is the index for highest temperature during the given duration?

(index = 0, 4, 5, 6) ✓

d. What is the index of lowest temperature during the given duration?

(index = 1, 7, 8) ✓

e. What day is the hottest day?

(Friday 21st, Tuesday 25th, Wednesday 26th, Thursday 27th) ✓

2. Trace the following code segment to determine the value stored in the `arr`.

a.
```
int [] arr = new int [8];
for (int i = 0; i < 8; i++) {
        arr[i] = i;
}
```

arr[0] = 0

behind the scene.

arr → nama array.

0	1	2	3	4	5	6	7
0	1	2	3	4	5	6	7

element dalam array merujuk kepada i.

0 1 2 3 4 5 6 7 ✓

b.
```
int [] arr = new int [8];
for (int i = 0; i < 8; i++) {
        arr[i] = i * i;
}
```

0 1 4 9 16 25 36 49 ✓

arr

0	1	2	3	4	5	6	7
0	1	4	9	16	25	36	49

c.
```
int [] arr = new int [8];
for (int i = 0; i < 8; i++) {
     if (i < 5)
          arr[i] = i * 2 +
 1;
     else
          arr[i] = i - 5;
}
```

1 3 5 7 9
0 1 2 ✓

arr

0	1	2	3	4	5	6	7
1	3	5	7	9	0	1	2

if | else

Section C: Write Code Segments

Full program, step by step.

1. Write code segments for the following task. Assume that the task will be executed in the given order.

a. Write Java statement that declare and create an array myList to hold 50 integers.

```
int [] = new int[50]
```

~~int [] myList = new int[];~~ `int [ ] myList = new int [50];`

b. Write code segments that reads 50 integers and store in array myList.

dah declare kat ca)

~~myList = new int [50];~~

```
for (int i = 0; i < 50; i++) {
 myList[i] = sc.nextInt();
 }
```

```
for (int i = 0 ; i<myList.length ; i++)
{
 myList[i] = sc.nextInt();
}
```

c. Write code segments to sum all elements in the array myList.

```
int sum = 0 ;
for (int i = 0; i < 50; i++)
{
 sum += myList[i];
} S.O.P in ( sum)
```

myList.length.

d. Write code segments to find the largest element in the array myList. max,min array boleh start dengan 1.

```
int max = 0;
for (int i = 0; i < n; i++)
{ (myList[i] > max)
  max = myList[i];
}
```

```
int max = myList [0];
for (int i = 0 ; i<myList.length ; i++)
{ if (myList[i] > max)
    max = myList[i];
}
S.O.P (max);
```

[ ] none need [ ]

e. Write code segments to find the index of the largest element in the array myList.

```
int indexOfMax = 0;
int max = 0;
for (int i = 0; i < n; i++)
{ if (myList[i] > max)
  { max = myList[i];
    indexOfMax = i;
  }
}
```

```
int indexMax = 0 ;
for(int i = 0 ; i<myList.length ; i++)
{ if (myList[i] > max)
    indexMax = i;
}
S.O.P (indexMax);
```

tak perlu [ ] sbb 0 tu index, kita bukan nak element dalam [ ].

f. Write code segments to copy all elements of array myList to array otherList.

```
1
2  public class CopyArray {
3
4      public static void main(String[] args) {
5
6          int[] firstArray = {5, 10, 15, 20, 25 };
7          int[] secondArray = new int[5];
8          for (int i = 0; i < firstArray.length; i++)
9          {
10             secondArray[i] = firstArray[i];
11             System.out.print(secondArray[i]+" ");
12         }
```

Console Problems Debug Shell

<terminated> CopyArray [Java Application] C:\Users\Tie\.p2\pool\plugins\org.ecl

5 10 15 20 25

① create otherList.

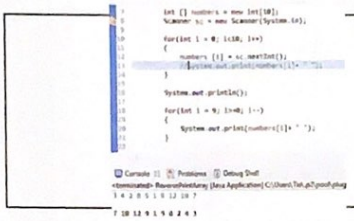
```
int [ ] myList = new int[50];
int [ ] otherList = new int [myList.length];
for (int i = 0; i<myList.length ; i++)
{ myList[i] = sc.nextInt();
  otherList[i] = myList[i];

  S.O.P (myList[i]);
  S.O.P (otherList[i]);

}
```

size refer myList.

* print kena dalam for.

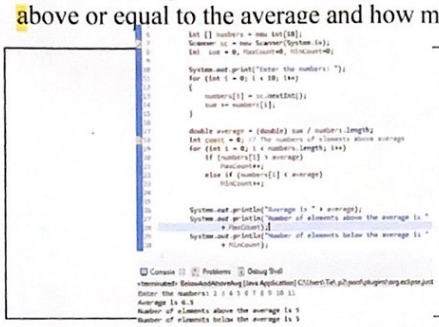
2. Write code segments that reads 10 integers between 1 and 10 and counts the occurrences of each.

3. Write code segments that reads 10 integers and displays them in reverse order. For example, if the data are 3 4 2 8 5 1 9 12 10 7, then the output is 7 10 12 9 1 5 8 2 4 3.



```
int [ ] numbers = new int [10]

for (int i=0 ; i<numbers.length ; i++)
{ numbers[i] = sc.nextInt(); }   looping u baca to input

for (int i = numbers.length-1; i>=0 ; i--)
{ S.O.P (numbers[i] + " "); }
```

4. Write code segments that reads 10 number of scores and determines how many scores are above or equal to the average and how many scores are below the average.



```
double average = 0 ;

for (int i=0; i<10 ; i++)
{ scores[i] = sc.nextInt();
  sum += scores[i] ;
}

average = sum / scores ;

int aboveAvg = 0 ;
int belowAvg = 0 ;
for (int i=0; i<10 ; i++)
{ if (scores[i] > average)
    aboveAvg ++ ;
  else if (scores[i] < average)
    belowAvg ++ ;
}
```

5. Write code segments that reads two lists of 10 integers and compare whether the two are strictly identical.

```
int[] firstArray = { 5, 10, 15, 20, 25 };
int[] secondArray = { 5, 10, 15, 20, 25 };
if (firstArray == secondArray)
  System.out.println("The arrays are the same.");
else
  System.out.println("The arrays are not the same.");
```

```
int [ ] num1 = new int[3];
int [ ] num2 = new int[3];
int count = 0;
for (int i=0; i<3; i++) {
    num1[i] = sc.nextInt();
    num2[i] = sc.nextInt();
}
for (int i=0 ; i<3; i++)
{ if (num1[i] == num2[i] )
    count++ ; }
if (count == 3 )
  S.O.P ("identical")
else
  S.O.P ("x identical")
```

6. Write code segments that reads 10 integers, remove the duplicate values in the array and display the result. For example; if the input is 1 2 3 2 1 6 3 4 5 2, then the output will be 1 2 3 6 4 5.

Code Segment

4.

```java
import java.util.Scanner;

public class BelowAndAboveAvg {
    public static void main(String[] args) {

        int [] numbers = new int[10];
        Scanner sc = new Scanner(System.in);
        int  sum = 0, MaxCount=0, MinCount=0;

        System.out.print("Enter the numbers: ");
        for (int i = 0; i < 10; i++)
        {
            numbers[i] = sc.nextInt();
            sum += numbers[i];
        }

        double average = (double) sum / numbers.length;
        int count = 0; // The numbers of elements above average
        for (int i = 0; i < numbers.length; i++)
            if (numbers[i] > average)
                MaxCount++;
            else if (numbers[i] < average)
                MinCount++;


        System.out.println("Average is " + average);
        System.out.println("Number of elements above the average is "
                + MaxCount);
        System.out.println("Number of elements below the average is "
                + MinCount);
```

Console Problems Debug Shell

<terminated> BelowAndAboveAvg [Java Application] C:\Users\Tie\.p2\pool\plugins\org.eclipse.justj.oper

```
Enter the numbers: 2 3 4 5 6 7 8 9 10 11
Average is 6.5
Number of elements above the average is 5
Number of elements below the average is 5
```

Tutorial 3

Tracing Table

2 b)

i	i < 8	i x i	i++
0	T	0	1
1	T	1	2
2	T	4	3
3	T	9	4
4	T	16	5
5	T	25	6
6	T	36	7
7	T	49	
8	F		

c)

i	i < 8	i < 5	i x 2 + 1	i - 5	i++
0	T	T	1		1
1	T	T	3		2
2	T	T	5		3
3	T	T	7		4
4	T	T	9		5
5	T	F		0	6
6	T	F		1	7
7	T	F		2	8
8	F				

d)

value [arr]	i	i < N (20)	arr[i-1] + arr[i-2]
arr[0] = 0			Fibonacci
[1] = 1			
arr[2] = 1	2	T	arr[1] + arr[0] = 0+1 = 1
arr[3] = 2	3	T	arr[2] + arr[1] = 1+1 = 2
arr[4] = 3	4	T	arr[3] + arr[2] = 2+1 = 3
arr[5] = 5	5	T	arr[4] + arr[3] = 3+2 = 5
arr[6] = 8	6	T	arr[5] + arr[4] = 5+3 = 8
arr[7] = 13	7	T	arr[6] + arr[5] = 8+5 = 13
arr[8] = 21	8	T	arr[7] + arr[6] = 13+8 = 21
arr[9] = 34	9	T	arr[8] + arr[7] = 21+13 = 34
arr[10] = 55	10	T	arr[9] + arr[8] = 34+21 = 55
arr[11] = 89	11	T	arr[10] + arr[9] = 55+34 = 89
arr[12] = 144	12	T	arr[11] + arr[10] = 89+55 = 144
arr[13] = 233	13	T	arr[12] + arr[11] = 144+89 = 233
arr[14] = 377	14	T	arr[13] + arr[12] = 233+144 = 377
arr[15] = 610	15	T	arr[14] + arr[13] = 377+233 = 610
arr[16] = 987	16	T	arr[15] + arr[14] = 610+377 = 987
arr[17] = 1597	17	T	arr[16] + arr[15] = 987+610 = 1597
arr[18] = 2584	18	T	arr[17] + arr[16] = 1597+987 = 2584
arr[19] = 4181	19	T	arr[18] + arr[17] = 2584+1597 = 4181
	20	F	