Arrays

COMP2026

PROBLEM SOLVING USING OBJECT ORIENTED PROGRAMMING

Storing a group of data

- In some situations, we need to keep track on **many data items** in the program for further processing
- Example: Write a program to read ten integers, then print the integers in reverse order
    - We may need to use ten **variables** to keep the ten integers in the program. It is rather troublesome

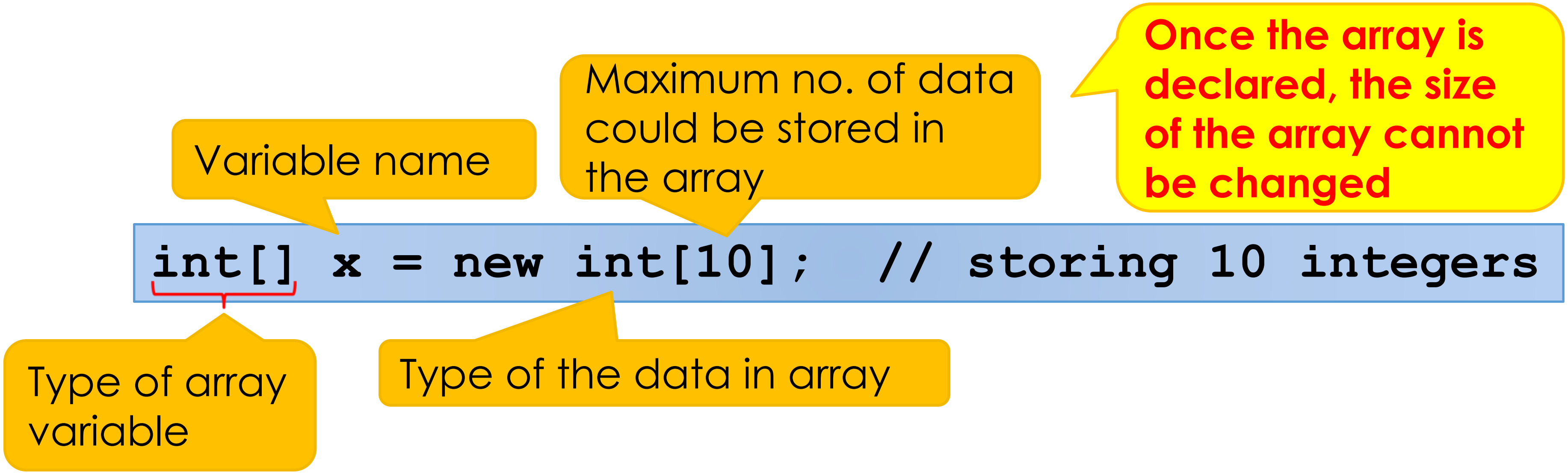
```
...
int x1, x2, x3, x4, x5, x6, x7, x8, x9, x10;
x1 = in.nextInt();
x2 = in.nextInt();
x3 = in.nextInt();
x4 = in.nextInt();
x5 = in.nextInt();
x6 = in.nextInt();
x7 = in.nextInt();
x8 = in.nextInt();
x9 = in.nextInt();
X10 = in.nextInt();

System.out.println(x10 + " " + x9 + " " + x8 +
" " + x7 + " " + x6 + " " + x5 + " " + x4 +
" " + x3 + " " + x2 + " " + x1);
...
```

It seems to be troublesome to use 10 variables for storing the 10 integers.

How about writing a program to read 100 numbers?? Create 100 variables?

Arrays

- An array is a **collection of variables** of the **same data type**
- To declare an array, we have to specify the type of the data and the maximum no. of data to be stored



```
int[] x = new int[10];  // storing 10 integers
```

Variable name

Maximum no. of data could be stored in the array

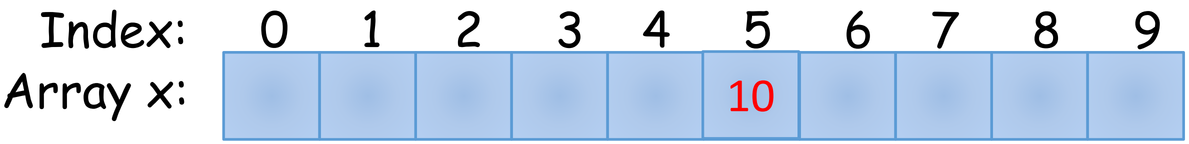
Once the array is declared, the size of the array cannot be changed

Type of array variable

Type of the data in array

Arrays

- An **array** is a group of **variables**
- Each **variable** is in a “slot” with an index



Index:	0	1	2	3	4	5	6	7	8	9
Array x:						10				

Index starts from 0. For an array of **10 items**, the **index is from 0 to 9**

- To access a value in an array , we specify which “slot” we want to use by using the **index operator [ ]**
- E.g. x[5]=10;

Put 10 into slot 5

Arrays

- Putting values into the array one by one

```
int[] x = new int[10];
x[0] = in.nextInt();
x[1] = in.nextInt();
x[2] = in.nextInt();
   ...
x[8] = in.nextInt();
x[9] = in.nextInt();
```

Not a good method!

Index:	0	1	2	3	4	5	6	7	8	9
Array x:	15	8	19	86	25	10	67	72	33	48

Arrays

- We could **use loop**

```
...
int[] x = new int[10];
for (int i = 0; i < x.length; i++){
  x[i] = in.nextInt();
}
```

Give us the size of array x, i.e. 10

i is looping from 0 to 9

Putting values into each slot of the array

Arrays

- Printing all the elements in an array

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

Give us the size of array x, i.e. 10

i is looping from 0 to 9

Arrays

- **Enhanced for loop**

For each item in an integer array x

```
...
for (int item : x){
  System.out.print(item + " ");
}
System.out.println();
...
```

Arrays

- **Enhanced for loop**

For each item in an integer array x

```
...
for (int item : x){
  System.out.print(item + " ");
}
System.out.println();
...
```

Array declaration with initial values

- When you declare an array, you can specify the initial values

List of initial values

```
int[] x = {12, 5, 27, 53, 5};
```

- Compare with

```
int[] x = new int[5];
```

Size of the array

- The size is omitted, and it is equal to the no. of items in the list

More examples on Array declaration

```
double[] scores = new double[100];
```

```
char[] grades = {'A', 'B', 'C', 'D', 'E'};
```

```
String[] names = new String[10];
```

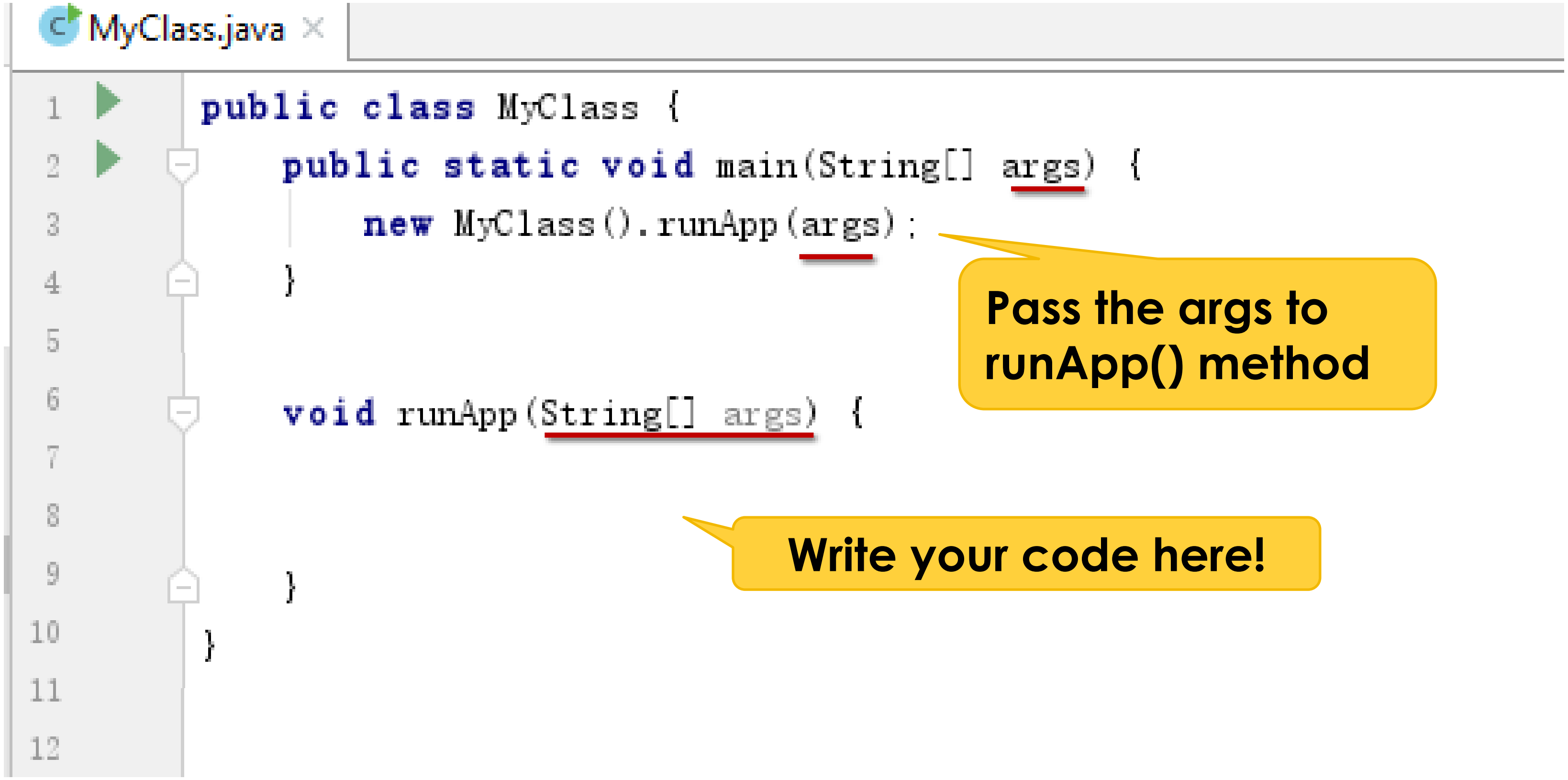
Part A
Discovery Exercises

Type your answer in **XXXXXXXX_lab04.docx**

Part B
Programming Exercises

Hints for Task 1

- Suppose you are told to write a Java program called MyClass to read command line arguments



```java
public class MyClass {
    public static void main(String[] args) {
        new MyClass().runApp(args);
    }

    void runApp(String[] args) {


    }
}
```

Pass the args to runApp() method

Write your code here!

Lab Exercise Submission

- Submit the following to Moodle
  - ***XXXXXXXX*_ lab04.docx**
  - ***XXXXXXXX*_lab04.zip**

*Replace “***XXXXXXXX***” with your **student ID**

Deadline: Next Monday 11:59 am

References

- Dean, J., & Dean, R. (2008). Introduction to programming with Java: A problem solving approach. Boston: McGraw-Hill.
- Forouzan, B. A., & Gilberg, R. F. (2007). Computer science: A structured programming approach using C (3rd ed.). Boston, MA: Thomson Course Technology.
- Gaddis, T. (2016). Starting out with Java (6th ed.). Pearson.
- Liang, Y. D. (2013). *Introduction to Java programming: Comprehensive version*. (8th ed.). Pearson.
- Schildt, H. (2006). *Java a beginner's guide*. New York: McGraw Hill.
- Wu, C. T. (2010). *An introduction to object-oriented programming with Java*. Boston: McGraw Hill Higher Education
- Xavier, C. (2011). Java programming: A practical approach. New Delhi: Tata McGraw Hill.
- Zakhour, S., Kannan, S., & Gallardo, R. (2013). The Java tutorial: A short course on the basics (5th ed.).
- yet another insignificant Programming Notes. (n.d.). Retrieved from https://www3.ntu.edu.sg/home/ehchua/programming