



Java Programming I

Session 3

Program Control Statements

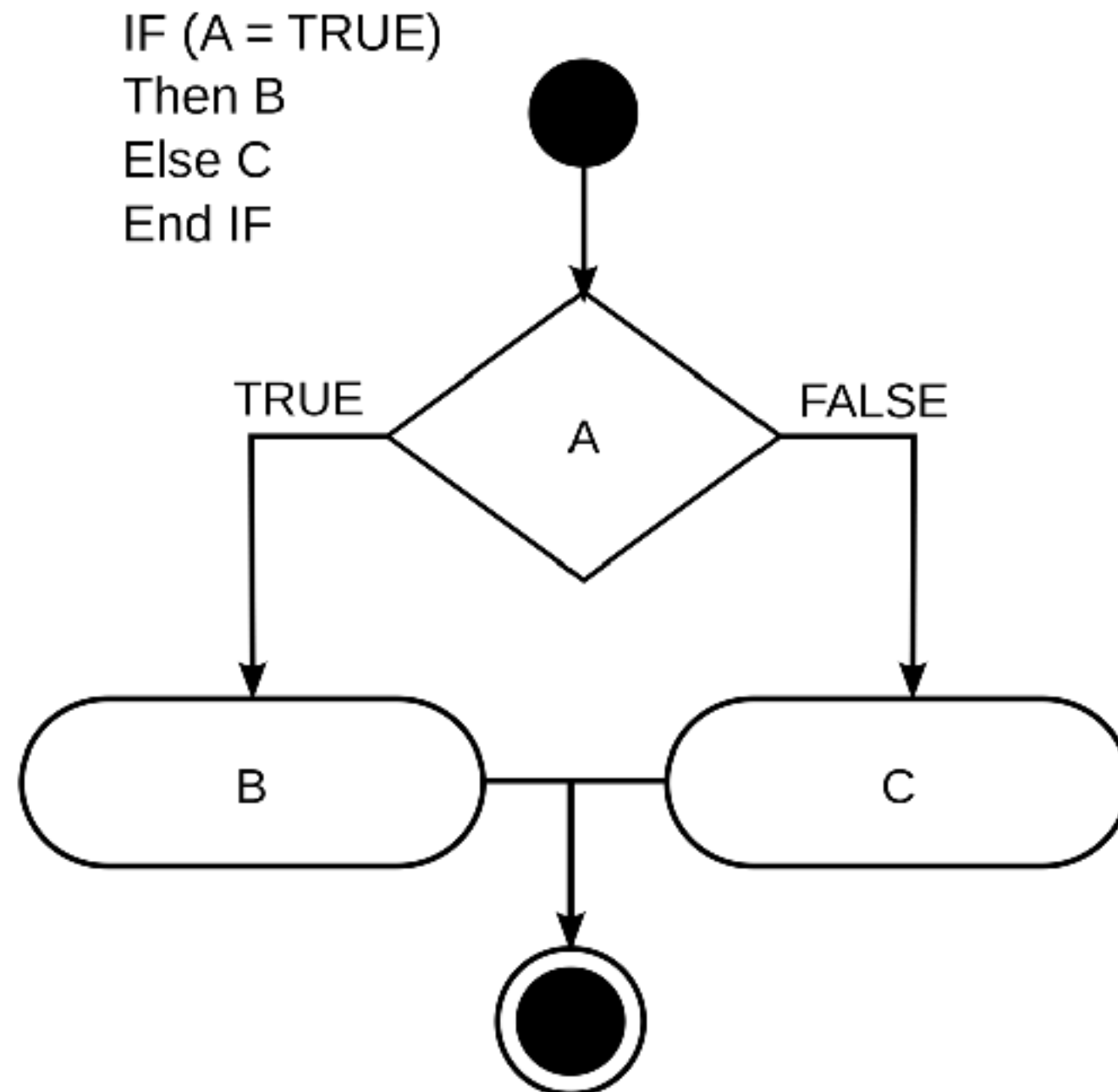
Juan Carlos Moreno - UCLA Ex

Agenda

- **IF/ELSE STATEMENT**
- **SWITCH STATEMENTS**
- **LOOPS**
- **ARRAYS**
- **STRINGS**

If & else

the basic conditional statement



If & else

chose between options

if (condition) statement

```
public class Product{
    int quantity;

    public boolean inStock(){
        if (quantity > 0){
            return true;
        }

        return false;
    }

    public static void main(String args[]){
        Product laptop_A = new Product();

        laptop_A.quantity = 10;

        if (laptop_A.inStock())
        {
            System.out.println("It is available!");
        } else {
            System.out.println("Sorry, it is unavailable!");
        }

        System.out.println("Continue");
    }
}
```

If, else if else

3 or more options

```
public class Product{
    int quantity;

    public static void main(String args[]){
        Product laptop_A = new Product();

        laptop_A.quantity = 5;

        if (laptop_A.quantity == 5)
        {
            System.out.println("We've got plenty");
        } else if (laptop_A.quantity == 4) {
            System.out.println("We're running out!");
        } else if (laptop_A.quantity == 3) {
            System.out.println("Almost gone!");
        } else {
            System.out.println("Act now!!");
        }

        System.out.println("Continue");
    }
}
```

Switch...case

one of many

```
switch (condition) {}
```

```
public class Product{
    int quantity;

    public static void main(String args[]){
        Product laptop_A = new Product();

        laptop_A.quantity = 5;

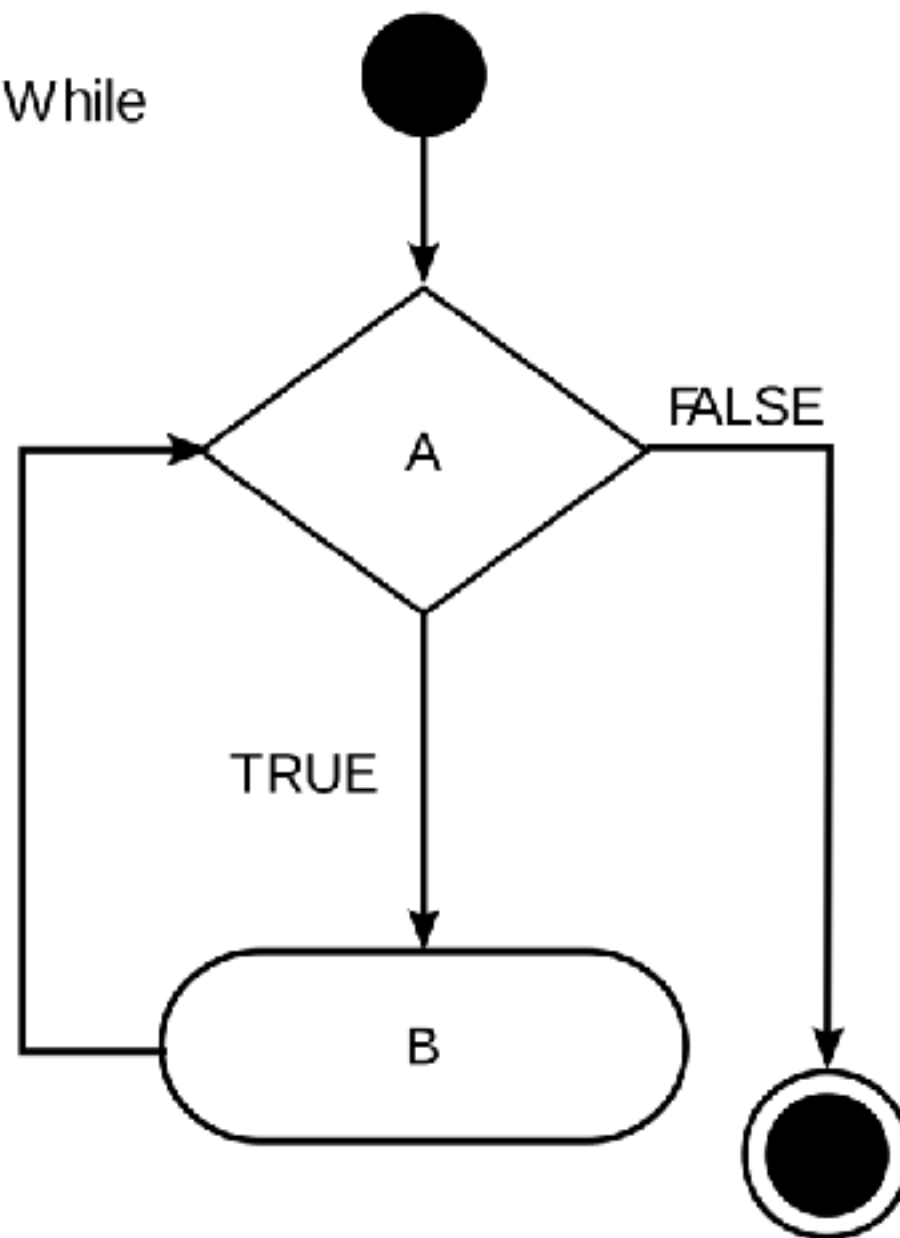
        switch (laptop_A.quantity){
            case 5:
                System.out.println("We've got plenty");
                break;
            case 4:
                System.out.println("We're running out!");
                break;
            case 3:
                System.out.println("Almost gone!");
                break;
            default:
                System.out.println("Act now!!");
        }

        System.out.println("Continue");
    }
}
```

Loops

rinse, wash, repeat

While (A= TRUE) Do
 B
End While



while...

not tied to a counter

```
int x = 0;
while (True){
    if (x > 10){
        break;
    }
    // Do something
    x++;
}
```


do while...

do at least once

```
int x = 0;

do {
    if (x > 10){
        break;
    }
    // Do something
    x++;
} while (True)
```

for

Object oriented no matter what

```
for (init;condition;increment)
```

```
for(int x = 0; x < 10; x++){  
    // Do something  
}
```

for (each)

Object oriented no matter what

```
for (type name: iterator)
```

```
int x[] = {1,2,3};
```

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

break

exit the loop

```
// Finding a multiple of 3
int x = 3;
while(true){
    int y = (int) (Math.random() * 100);

    if (y % x == 0)
    {
        System.out.println("Found y:" + y + " and x:" + x);
        break;
    }
}
```

continue

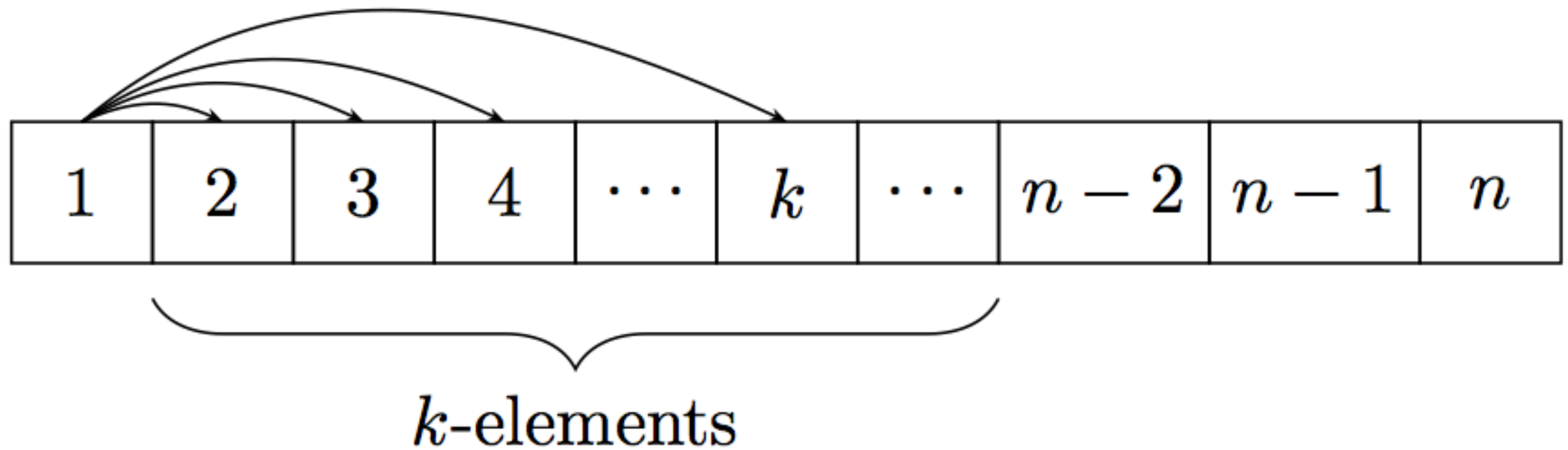
useful for performance

```
// Once done, don't even process the rest
for(int x = 2; x < 10; x++){
    int y = (int) (Math.random() * 100);

    if (y % x == 0)
    {
        System.out.println("Found y:" + y + " and x:" + x);
        break;
    }
    continue;
}
```

Arrays

a list of elements

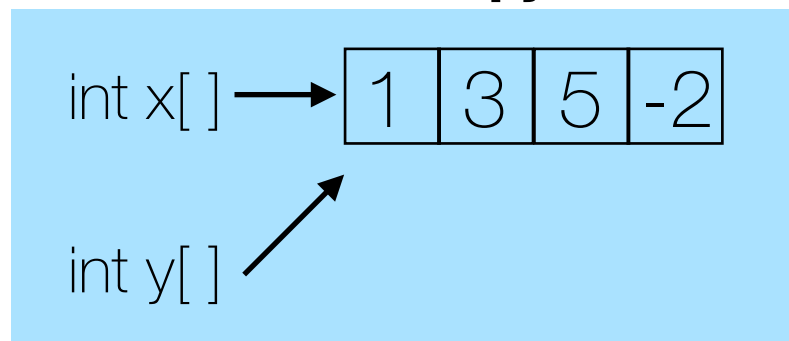


Arrays

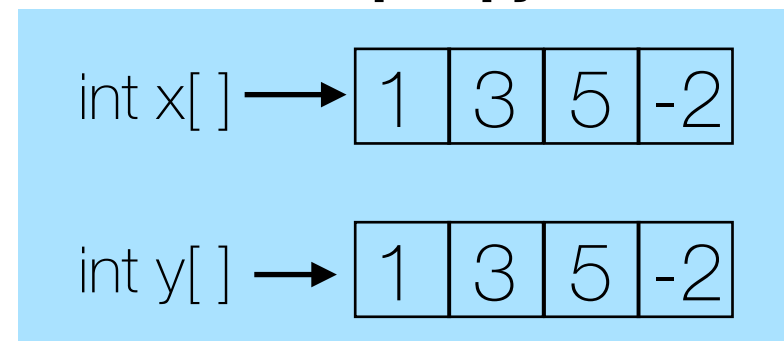
Special considerations

- Once declared, they cannot change in size
- All elements are one single data type
- Created using a creator (new) or initializer
- Item values have no particular order

shallow copy



deep copy



Arrays

Read operation

```
int ages[] = {20,22,23,28,32};  
  
for (int x=0; x<ages.length; x++){  
    String msg = String.format("Person #%d = %d",x, ages[x]);  
    System.out.println(msg);  
}
```


Arrays

Write Operation

```
int ages[]=new int[5];
int max_age = 30;

for (int x=0; x<ages.length; x++){
    int random_age = ((int) (Math.random()*100) % max_age);
    System.out.println(random_age);
    ages[x] = random_age;
}

String msg = String.format("All ages: %s",Arrays.toString(ages));
System.out.println(msg);
```

Arrays

Sort the following

```
int[] ages= {24, 36, 14, 35, 22, 37, 42, 25};
```

Strings

a list of characters

index	0	1	2	3	4	5
String x	H	e	l	l	o	\0

Strings

Special considerations

- They are char arrays. All array considerations apply
- Strings are immutable
- For convenience java provides String object vs char[]
- Strings work with Switch statements
- Strings are not a primitive ("string" == String object)

Strings

String exercises

```
char data[] = {'R','i','n','g','o'}; // No double quotes for chars
String drummer = new String(data);
String drummer2 = "Rin" + "go";
if(drummer == drummer2)
{
    System.out.println("Both are ");
}

System.out.println(drummer + " vs " + drummer2);
```

Then

- Compare them using == vs equals method
- Try using the API to replace the R for B
- Try using the API to find out if the string starts with “Bing”
- Try using the API to get the last 4 characters as a string
- Try using the API to remove trailing spaces

Strings

Reverse the following String

```
String title = "Strawberry Fields Forever";
```