CSE 20
Intro to Computing I

Lecture 8 – Loop Control Flow (2)

Arrays (1)

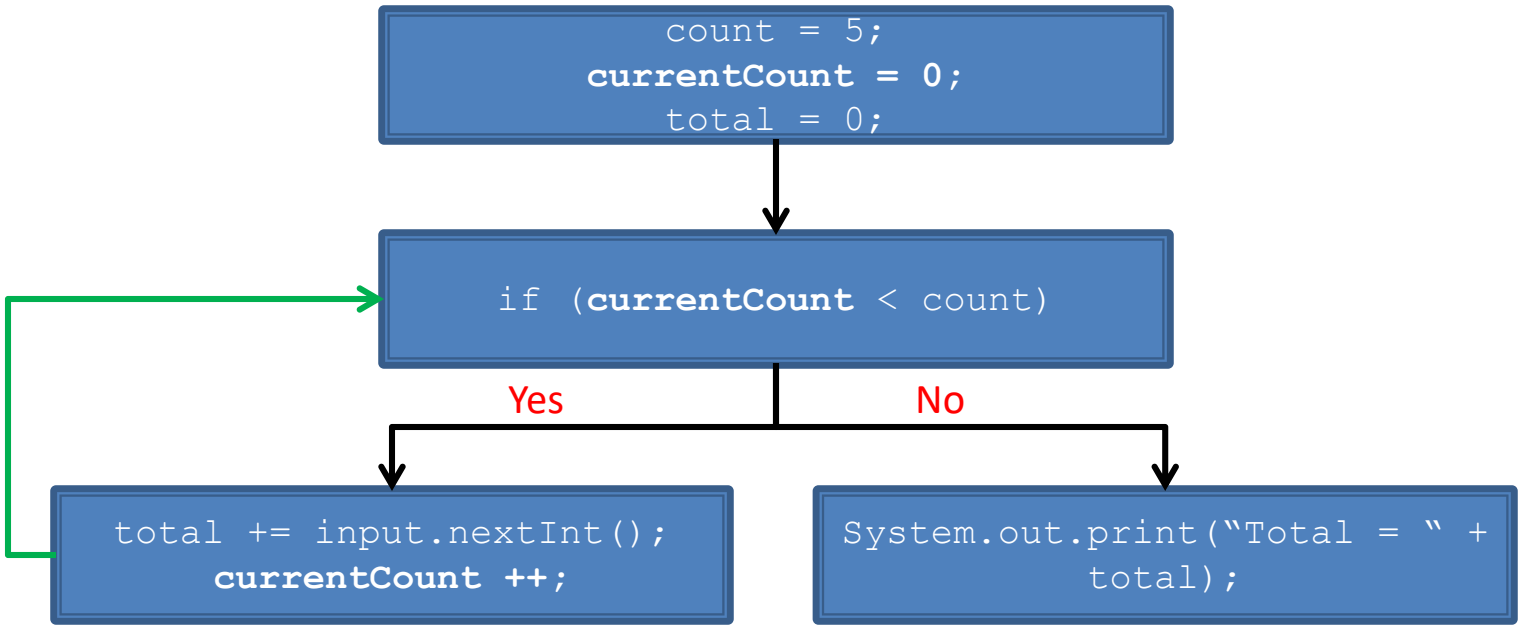
Announcements

- Lab #10 this week
  - Due before your next lab
  - Make sure to show your work to **YOUR TA** (or me) before submission
- Project #2 out this Friday
  - Due 12/1 (Friday)
- Reading assignment
  - Chapter 5.1-5.5 of textbook (Due date extended)

Check your UC Merced email for your invitation to take the New Student Survey!

- All new first-year and transfer students received an email invitation
  - Or visit: http://irds.ucmerced.edu/new_student_survey.html
- The survey…
  - Only takes about 10 minutes and ends November 20th
  - Asks about your educational plans, your transition to UCM, and the UCM services you use
- Your feedback will help us…
  - Improve the student experience for new undergraduates
  - Better understand what we can do to help you be academically successful

Sum of 5 Numbers (Loop)



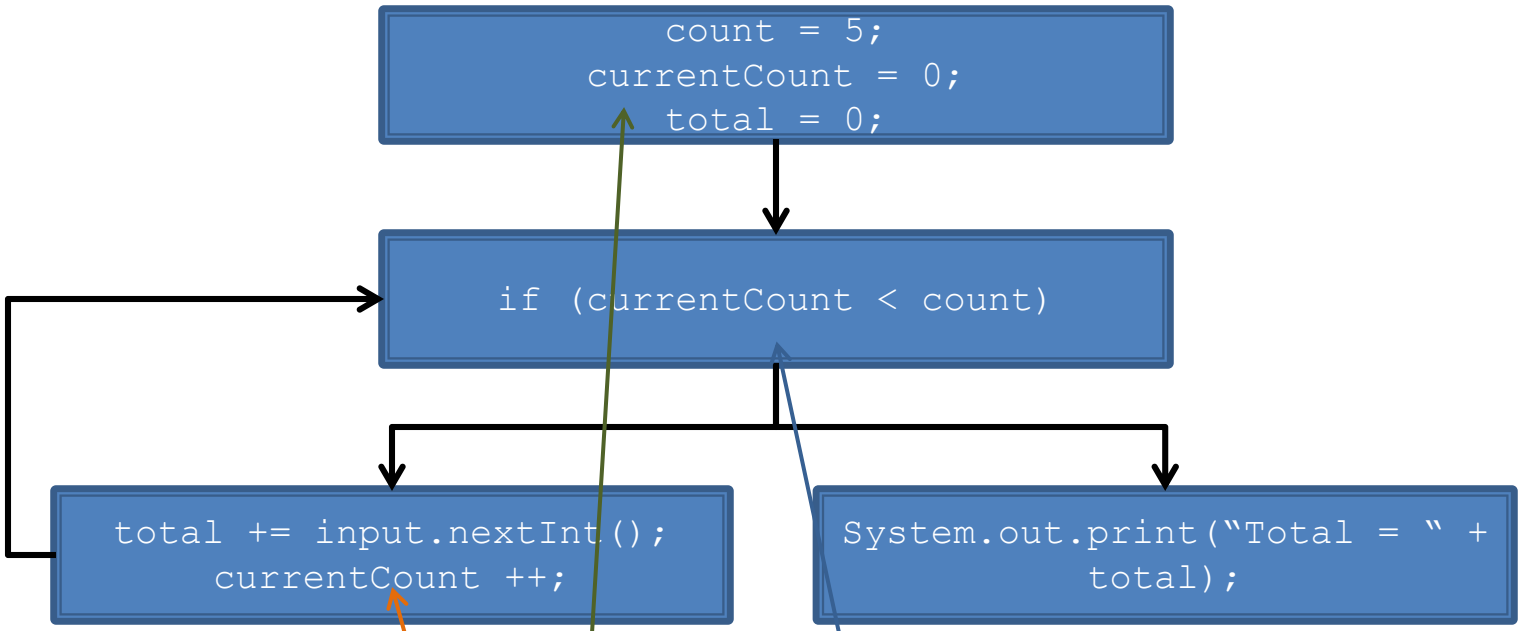
While-Loop

```
while ( running condition ) {
    …
    body
}
```

For-Loop

```
for ( initialization; continue condition; increment expression ) {
        …
        body
}
```

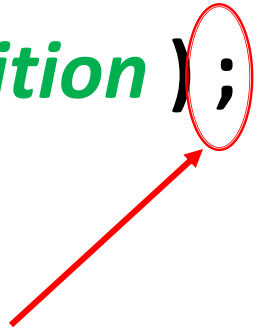
Do-while Loop




```
count = 5;
total = 0;
currentCount = 0;
do {
     total += input.nextInt();
     currentCount++;
} while (currentCount < count);
System.out.print("Total = " + total);
```

Do-while-Loop

```
do{
    …
    body
} while ( running condition ) ;
```



There is a ‘ ; ’ at the end!

Example : AllNumDoWhile.java

```
System.out.print("Please enter the max number:");

int max = input.nextInt();
int i = 0;
do
{
  System.out.println("Number " + i);
  i++;
}while (i <= max);
```

```
Please enter the max number:5
Number 0
Number 1
Number 2
Number 3
Number 4
Number 5
```

Example : AllNumDoWhile.java

```
System.out.print("Please enter the max number:");

int max = input.nextInt();
int i = 0;
do
{
  System.out.println("Number " + i);
  i++;
}while (i <= max);
```

Please enter the max number:-1
Number 0

Body of loop will run at least one time!

Example : If-Else Nested

```
System.out.print("Please enter the max number:");

int max = input.nextInt();
int i = 0;
do {
 if (i == 2)
  System.out.println("Two");
 else
  System.out.println("Number " + i);
  i = i + 1; // i++
}while (i <= max);
```

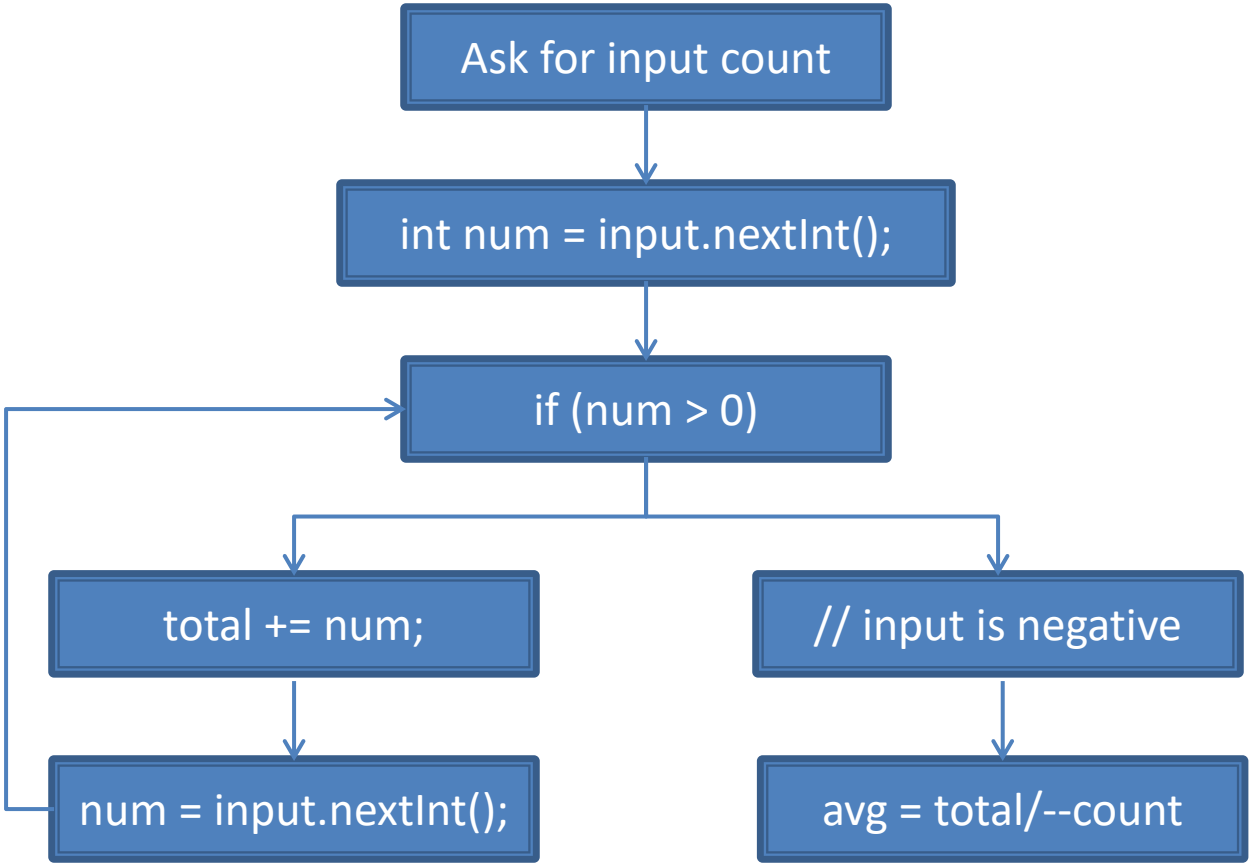
```
Please enter the max number:5
Number 0
Number 1
Two
Number 3
Number 4
Number 5
```

Break: end early

```
System.out.print("Please enter the max number:");
int max = input.nextInt();
int i = 0;
do {
 if (i == 2)
   break;
 System.out.println("Number " + i);
 i++;
}while (i <= max);
```

Please enter the max number: 5
Number 0
Number 1

PosAverage.java



While Loop

```
public static void main(String[] args) {
 int count = 0;
 int total = 0;

 Scanner input = new Scanner(System.in);

 System.out.print("Enter " + count++ + " number: ");
 int num = input.nextInt();

 while (num > 0) {
  total += num;
  System.out.print("Enter " + count++ + " number: ");
  num = input.nextInt();
 }

 System.out.println("Average is " + total/(--count));
}
```

There is an extra count after exiting the loop

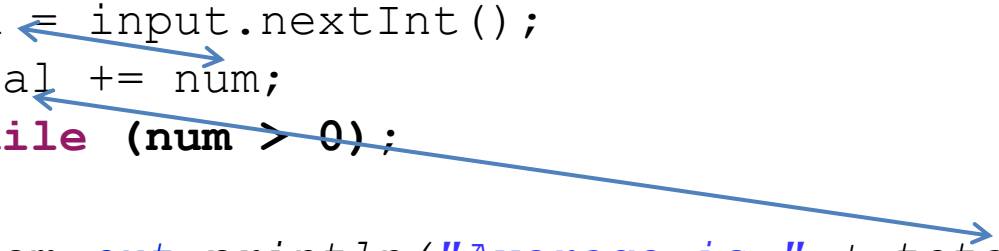
Do-While #1

```
public static void main(String[] args) {
 int count = 0;
 int total = 0;

 Scanner input = new Scanner(System.in);

 do {
  System.out.print("Enter " + count++ + " number: ");
  int num = input.nextInt();
  total += num;
 } while (num > 0);

 System.out.println("Average is " + total/--count);
}
```

Re-defining num inside the loop

Do-While #2

```
public static void main(String[] args) {
 int count = 0;
 int total = 0;

 Scanner input = new Scanner(System.in);
 int num;
 do {
  System.out.print("Enter " + count++ + " number: ");
  num = input.nextInt();
  total += num;
 } while (num > 0);

 System.out.println("Average is " + total/--count);
}
```



What if the 1st num is negative?

Do-While #3

```
public static void main(String[] args) {
 int count = 0;
 int total = 0;

Scanner input = new Scanner(System.in);
 int num = 0;
 do {
  total += num;
  System.out.print("Enter " + count++ + " number: ");
  num = input.nextInt();
 } while (num > 0);

 System.out.println("Average is " + total/--count);
}
```

Do-While : Final

```
public static void main(String[] args) {
 int count = 0;
 int total = 0;

 Scanner input = new Scanner(System.in);
 int num = 0;
 do {
  total += num;
  System.out.print("Enter " + count++ + " number: ");
 } while ((num = input.nextInt()) > 0);

 System.out.println("Average is " + total/--count);
}
```

Variable Dependency in Loops

- For
  - All related
- While/do-while
  - Does not have to
  - It can be related

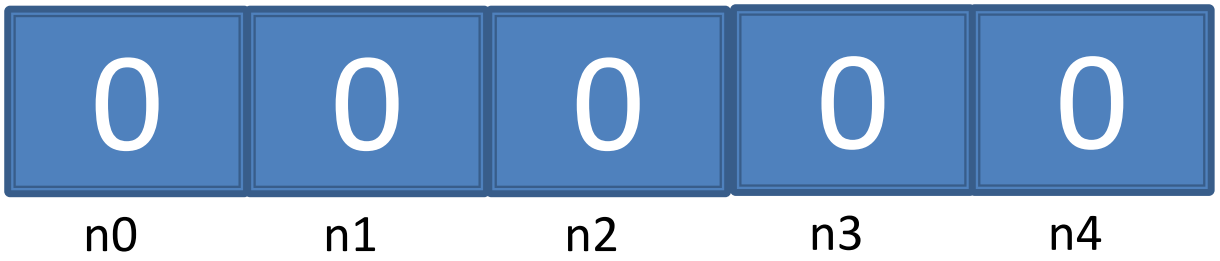
Loops - Summary

- For
  - Pre-determined number of iterations
  - Variable use is structured
    - initialization ; continue condition; increment
- While
  - Most general form
  - Can iterate any amount of times
  - Does not need a variable
- Do-While
  - Does at least one iteration
  - Rest is same as a while-loop **(extra ;)**

Variables

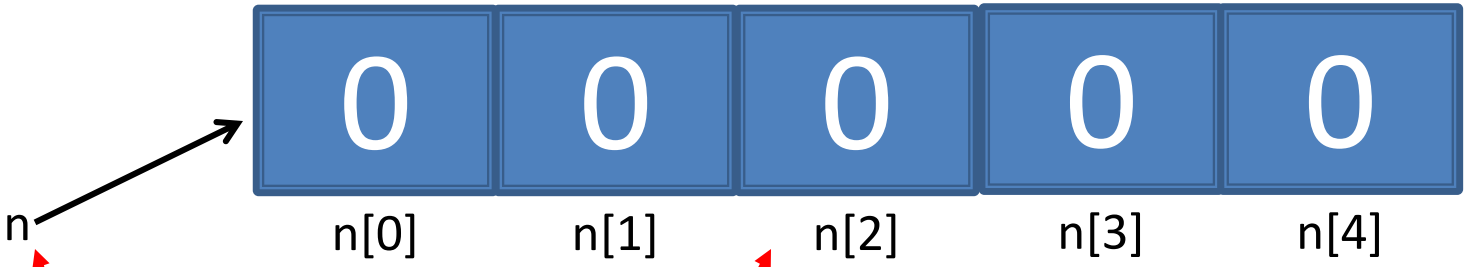
- int total = 0;

- int n0 = 0, n1 = 0, n2 = 0, n3 = 0, n4 = 0;



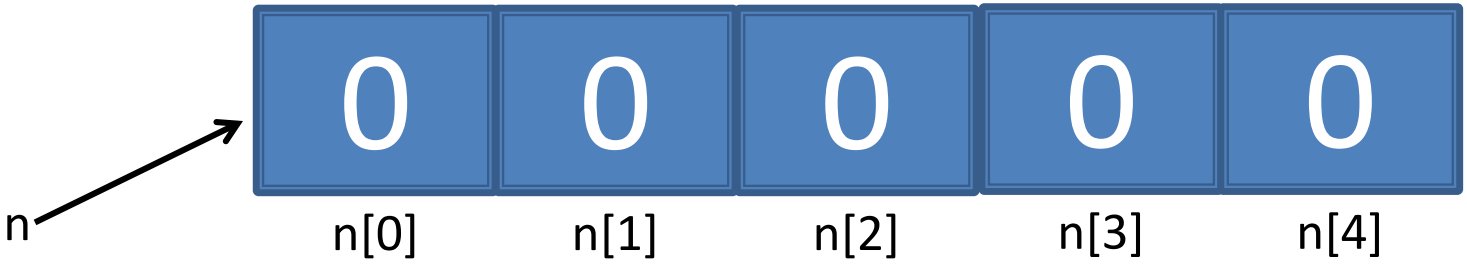
- What will happen if there are 1000 numbers?
- int n5 = 0, ... n999 = 0;
- Not very efficient!

Array - Initialization



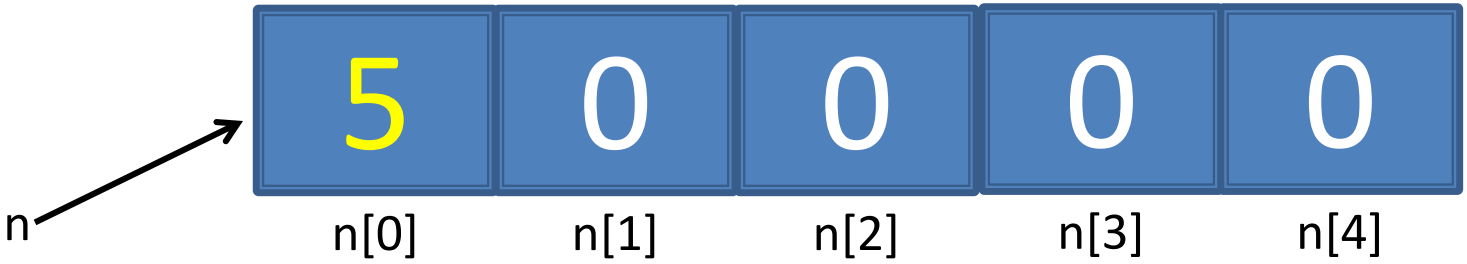
- Creates 5 spaces of integer (variable n)
- [ ] denotes an array OR the index
  - two different meanings
- Declaration:
  - int[] n = new int[5];
  - int[] n = {0, 0, 0, 0, 0};

Array - Declaration



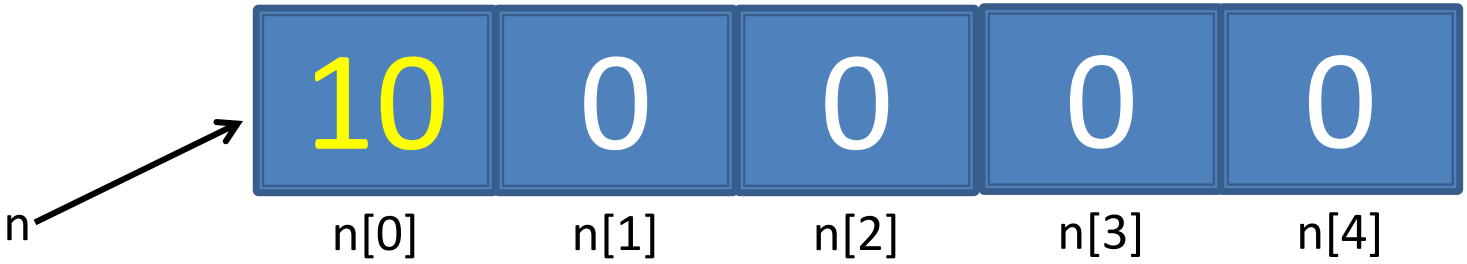
- Create a pointer first
  - int [] n;
- Create memory space
  - n = new int[5];

Array - Usage



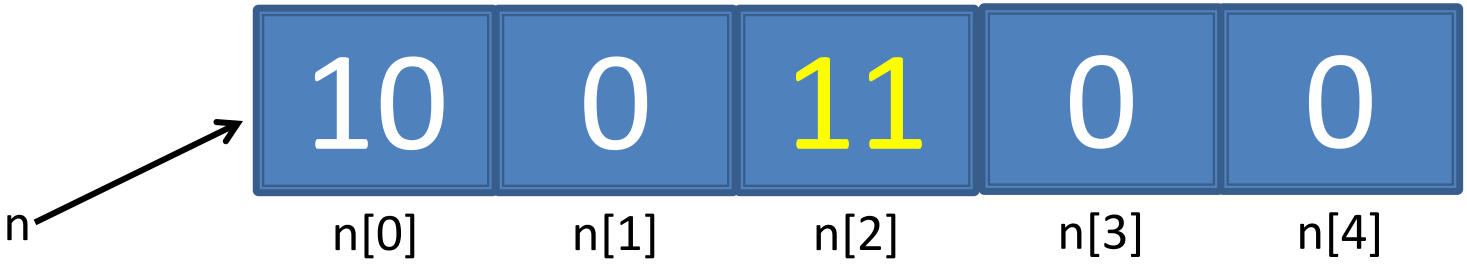
- n[0] = 5;
- Using the value inside the [ ] to access the content.

Array - Usage



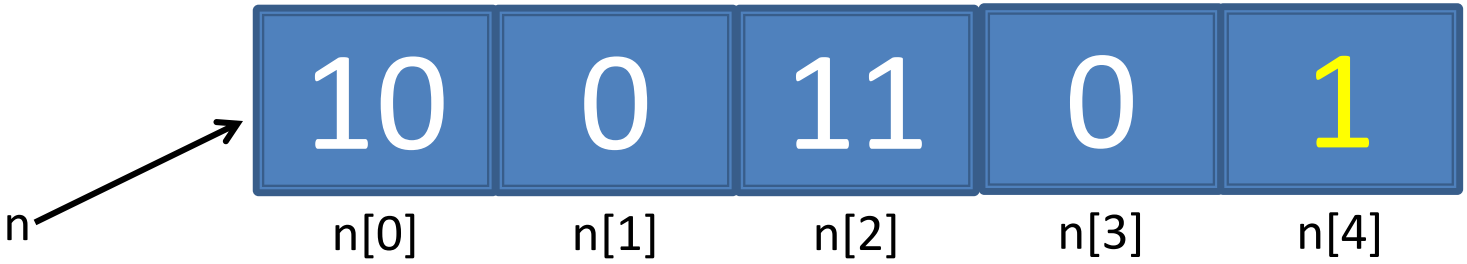
- n[0] = 5;
- n[0] += 5;

Array - Usage



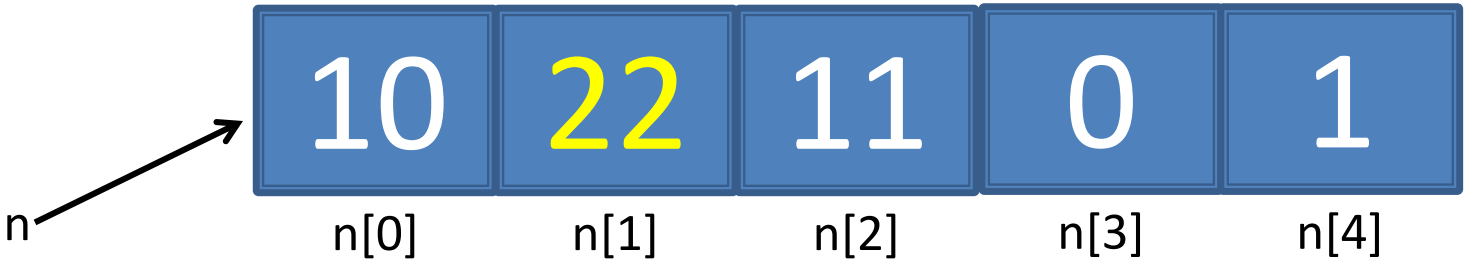
- n[0] = 5;
- n[0] += 5;
- n[2] = n[0] + 1;

Array - Usage



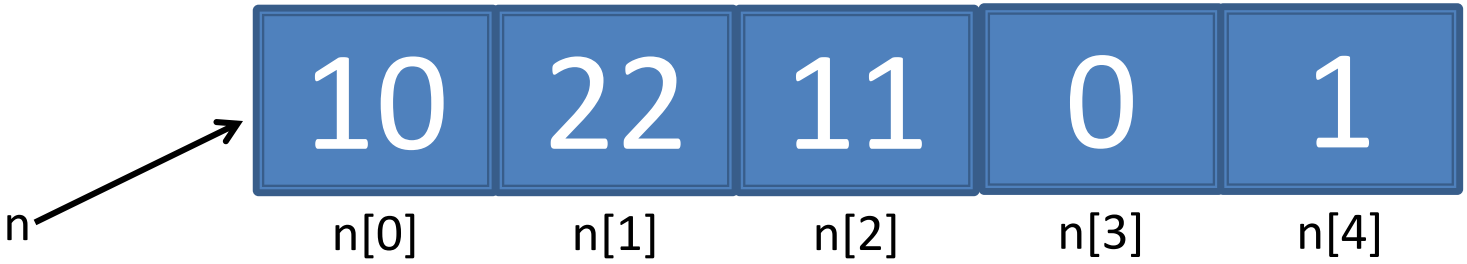
- n[0] = 5;
- n[0] += 5;
- n[2] = n[0] + 1;
- n[4] += 1;

Array - Usage



- n[0] = 5;
- n[0] += 5;
- n[2] = n[0] + 1;
- n[4] += 1;
- n[1] = n[0] + n[1] + n[2] + n[3] + n[4];

Array - Usage




```
for (int i = 0; i < n.length; i++)
    System.out.println("Index " + i + " has value " + n[i]);
```

Index 0 has value 10
Index 1 has value 22
Index 2 has value 11
Index 3 has value 0
Index 4 has value 1

n.Length tells you have many elements are in n

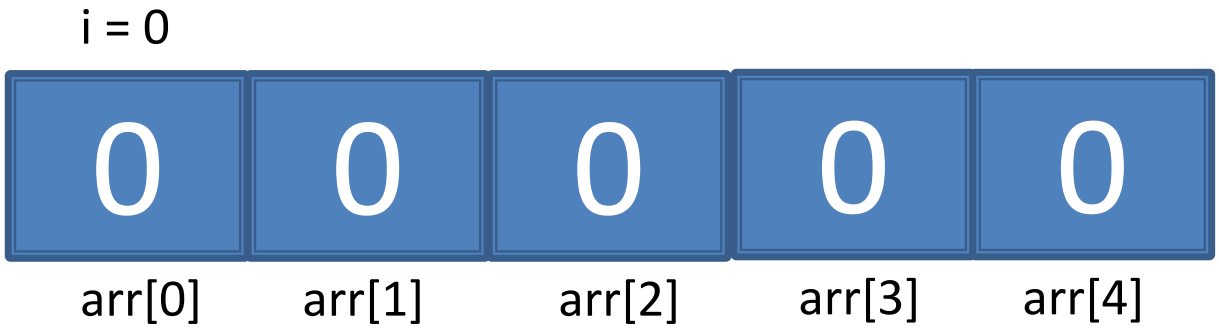
Array : Pit Falls

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

for (int i = 0; i <= arr.length; i++)
 System.out.println("Index " + i + " is " + arr[i]);
}
```

What is wrong?

Array : Pit Falls

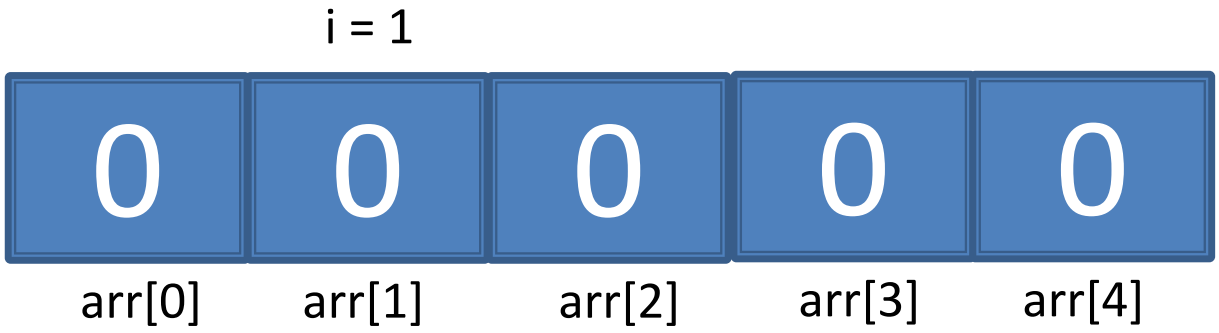
```
int[] arr = new int[5];

for (int i = 0; i <= arr.length; i++)
 System.out.println("Index " + i + " is " + arr[i]);
}
```



Index 0 is 0

Array : Pit Falls

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

for (int i = 0; i <= arr.length; i++)
 System.out.println("Index " + i + " is " + arr[i]);
}
```



Index 0 is 0
Index 1 is 0

Array : Pit Falls

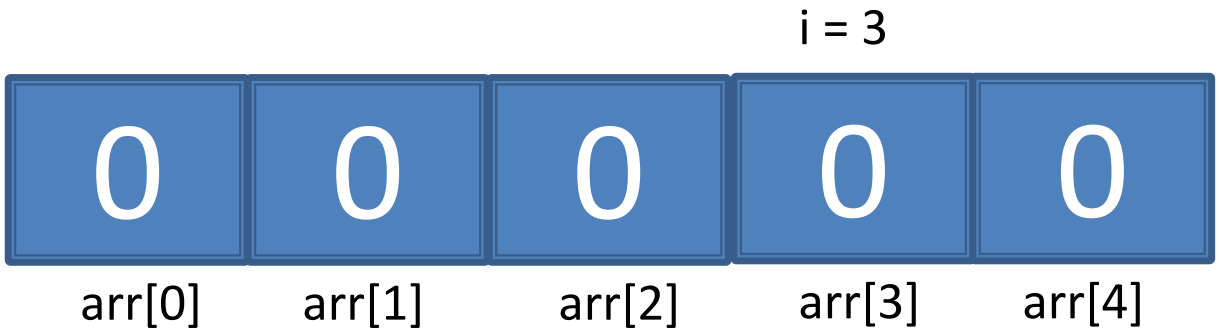
```
int[] arr = new int[5];

for (int i = 0; i <= arr.length; i++)
 System.out.println("Index " + i + " is " + arr[i]);
}
```

i = 2

Index 0 is 0
Index 1 is 0
Index 2 is 0

Array : Pit Falls

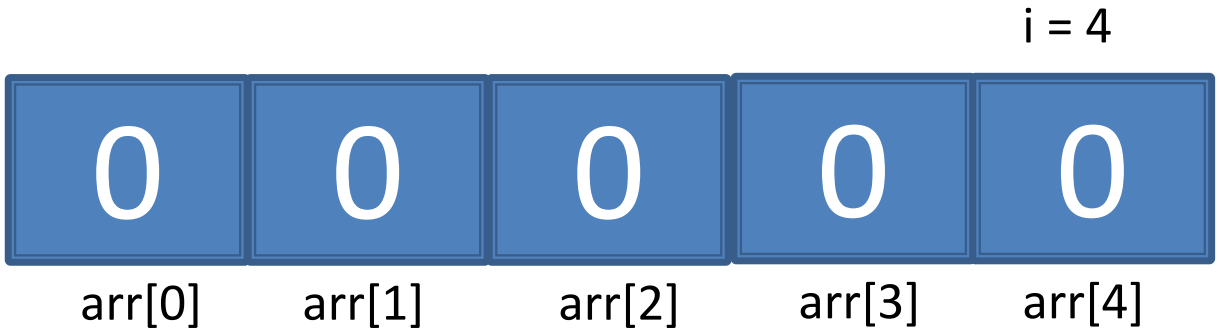
```
int[] arr = new int[5];

for (int i = 0; i <= arr.length; i++)
 System.out.println("Index " + i + " is " + arr[i]);
}
```



Index 0 is 0
Index 1 is 0
Index 2 is 0
Index 3 is 0

Array : Pit Falls

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

for (int i = 0; i <= arr.length; i++)
 System.out.println("Index " + i + " is " + arr[i]);
}
```



Index 0 is 0
Index 1 is 0
Index 2 is 0
Index 3 is 0
Index 4 is 0

Array : Pit Falls

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

for (int i = 0; i <= arr.length; i++)
 System.out.println("Index " + i + " is " + arr[i]);
}
```

Logical Error

i = 5


```
Exception in thread "main"
java.lang.ArrayIndexOutOfBoundsException: 5
at ErrorsTest.main(ErrorsTest.java:7)
```

Runtime Error (i = 5 but arr[5] does not exist)