


Passaggio di tipi primitivi

```
public static void incrementa(int x) {  
    x++;  
}
```

```
public static void main(String[] s) {  
    int a = 3;  
    System.out.println(a);  
    incrementa(a);  
    System.out.println(a);  
}
```

a 

Passaggio di tipi primitivi

```
public static void incrementa(int x) {  
    x++;  
}
```

```
public static void main(String[] s) {  
    int a = 3;  
    System.out.println(a);  
    incrementa(a);  
    System.out.println(a);  
}
```

a

3

- valuto parametro attuale (valore: 3)

Passaggio di tipi primitivi

```
public static void incrementa(int x) {  
    x++;  
}
```

```
public static void main(String[] s) {  
    int a = 3;  
    System.out.println(a);  
    incrementa(a);  
    System.out.println(a);  
}
```

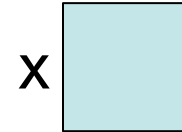
a

3

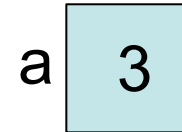
- valuto parametro attuale (valore: 3)
- creo parametro formale (la variabile x locale al metodo incrementa)

Passaggio di tipi primitivi

```
public static void incrementa(int x) {  
    x++;  
}
```



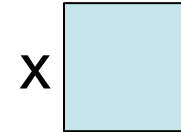
```
public static void main(String[] s) {  
    int a = 3;  
    System.out.println(a);  
    incrementa(a);  
    System.out.println(a);  
}
```



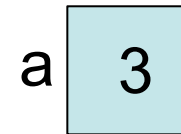
- valuto parametro attuale (valore: 3)
- creo parametro formale (la variabile x locale al metodo incrementa)

Passaggio di tipi primitivi

```
public static void incrementa(int x) {  
    x++;  
}
```



```
public static void main(String[] s) {  
    int a = 3;  
    System.out.println(a);  
    incrementa(a);  
    System.out.println(a);  
}
```



- valuto parametro attuale (valore: 3)
- creo parametro formale (la variabile x locale al metodo incrementa)
- inizializzo il parametro formale (x, contiene il valore 3)

Passaggio di tipi primitivi

```
public static void incrementa(int x) {  
    x++;  
}
```

x 3


```
public static void main(String[] s) {  
    int a = 3;  
    System.out.println(a);  
    incrementa(a);  
    System.out.println(a);  
}
```

a 3

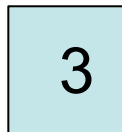
- valuto parametro attuale (valore: 3)
- creo parametro formale (la variabile x locale al metodo incrementa)
- inizializzo il parametro formale (x, contiene il valore 3)

Passaggio di tipi primitivi

```
public static void incrementa(int x) {  
    x++;  
}
```

x 


```
public static void main(String[] s) {  
    int a = 3;  
    System.out.println(a);  
    incrementa(a);  
    System.out.println(a);  
}
```

a 

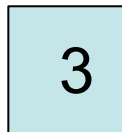
- valuto parametro attuale (valore: 3)
- creo parametro formale (la variabile x locale al metodo incrementa)
- inizializzo il parametro formale (x, contiene il valore 3)
- eseguo il metodo (incrementa la variabile x a 4, a resta immutata)

Passaggio di tipi primitivi

```
public static void incrementa(int x) {  
    x++;  
}
```

x 


```
public static void main(String[] s) {  
    int a = 3;  
    System.out.println(a);  
    incrementa(a);  
    System.out.println(a);  
}
```

a 

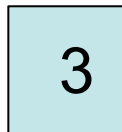
- valuto parametro attuale (valore: 3)
- creo parametro formale (la variabile x locale al metodo incrementa)
- inizializzo il parametro formale (x, contiene il valore 3)
- eseguo il metodo (incrementa la variabile x a 4, a resta immutata)

Passaggio di tipi primitivi

```
public static void incrementa(int x) {  
    x++;  
}
```

x 

```
public static void main(String[] s) {  
    int a = 3;  
    System.out.println(a);  
    incrementa(a);  
    System.out.println(a);  
}
```

a 

- valuto parametro attuale (valore: 3)
- creo parametro formale (la variabile x locale al metodo incrementa)
- inizializzo il parametro formale (x, contiene il valore 3)
- eseguo il metodo (incrementa la variabile x a 4, a resta immutata)
- al termine del metodo le variabili locali vengono eliminate

Passaggio di tipi primitivi

```
public static void incrementa(int x) {  
    x++;  
}
```

```
public static void main(String[] s) {  
    int a = 3;  
    System.out.println(a);  
    incrementa(a);  
    System.out.println(a);  
}
```

a

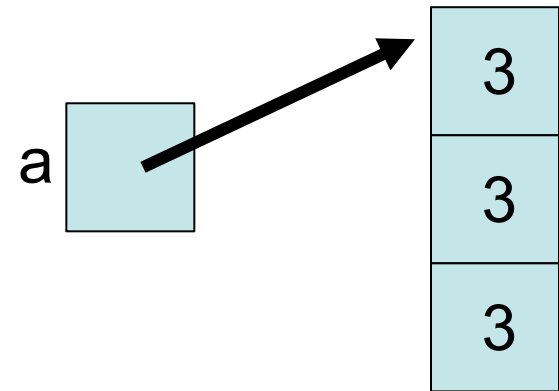
3

- valuto parametro attuale (valore: 3)
- creo parametro formale (la variabile x locale al metodo incrementa)
- inizializzo il parametro formale (x, contiene il valore 3)
- eseguo il metodo (incrementa la variabile x a 4, a resta immutata)
- al termine del metodo le variabili locali vengono eliminate

Passaggio di riferimenti

```
public static void incrementa(int[] x) {  
    for (int i=0; i<x.length; i++)  
        x[i]++;  
}
```

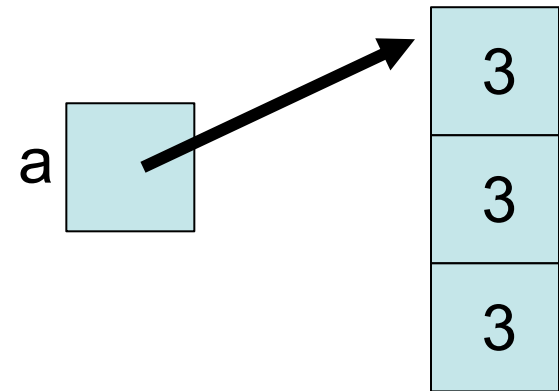
```
public static void main(String[] s) {  
    int[] a = {3, 3, 3};  
    stampaVett(a);  
    incrementa(a);  
    stampaVett(a);  
}
```



Passaggio di riferimenti

```
public static void incrementa(int[] x) {  
    for (int i=0; i<x.length; i++)  
        x[i]++;  
}
```

```
public static void main(String[] s) {  
    int[] a = {3, 3, 3};  
    stampaVett(a);  
    incrementa(a);  
    stampaVett(a);  
}
```

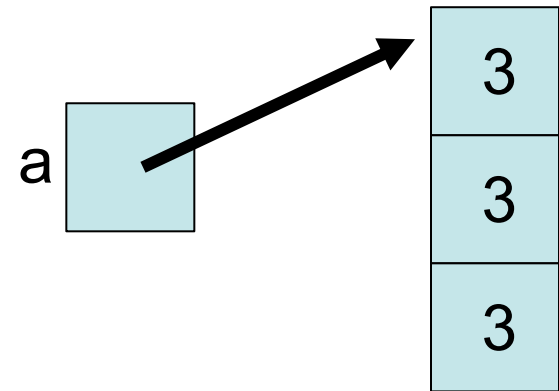


- valuto parametro attuale (valore: riferimento all'array {3, 3, 3})

Passaggio di riferimenti

```
public static void incrementa(int[] x) {  
    for (int i=0; i<x.length; i++)  
        x[i]++;  
}
```

```
public static void main(String[] s) {  
    int[] a = {3, 3, 3};  
    stampaVett(a);  
    incrementa(a);  
    stampaVett(a);  
}
```

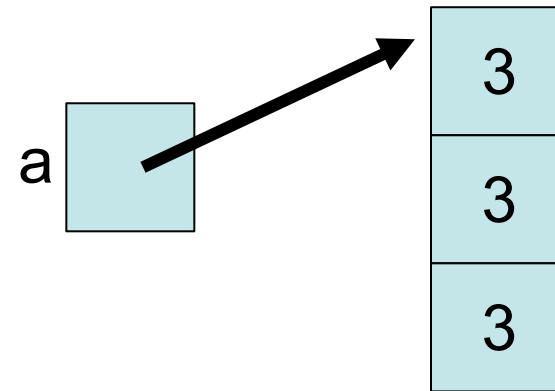
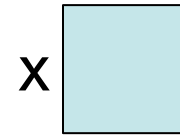


- valuto parametro attuale (valore: riferimento all'array {3, 3, 3})
- creo parametro formale (riferimento x)

Passaggio di riferimenti

```
public static void incrementa(int[] x) {  
    for (int i=0; i<x.length; i++)  
        x[i]++;  
}
```

```
public static void main(String[] s) {  
    int[] a = {3, 3, 3};  
    stampaVett(a);  
    incrementa(a);  
    stampaVett(a);  
}
```

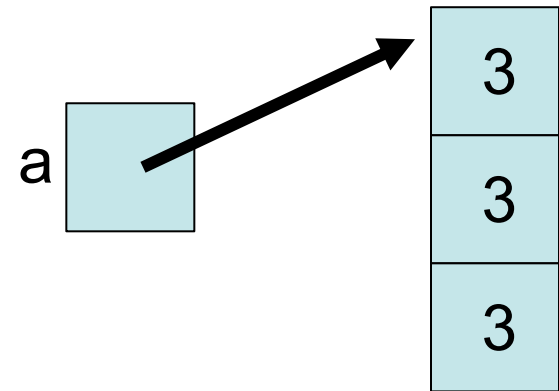
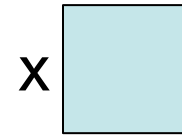


- valuto parametro attuale (valore: riferimento all'array {3, 3, 3})
- creo parametro formale (riferimento x)

Passaggio di riferimenti

```
public static void incrementa(int[] x) {  
    for (int i=0; i<x.length; i++)  
        x[i]++;  
}
```

```
public static void main(String[] s) {  
    int[] a = {3, 3, 3};  
    stampaVett(a);  
    incrementa(a);  
    stampaVett(a);  
}
```

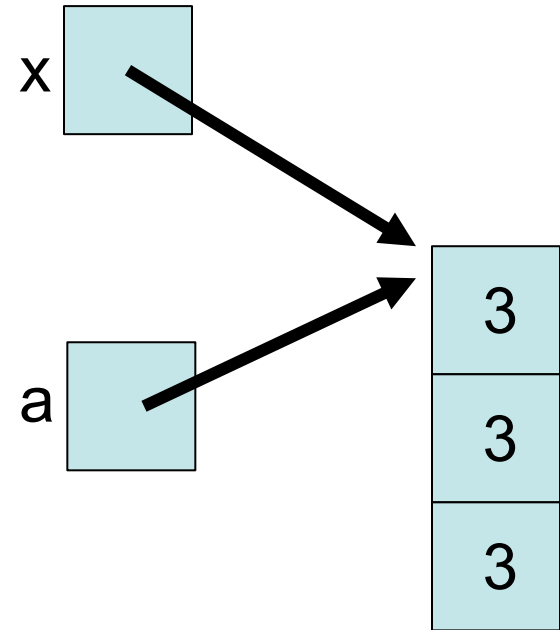


- valuto parametro attuale (valore: riferimento all'array {3, 3, 3})
- creo parametro formale (riferimento x)
- inizializzo il parametro formale (x ora è un riferimento all'array {3, 3, 3})

Passaggio di riferimenti

```
public static void incrementa(int[] x) {  
    for (int i=0; i<x.length; i++)  
        x[i]++;  
}
```

```
public static void main(String[] s) {  
    int[] a = {3, 3, 3};  
    stampaVett(a);  
    incrementa(a);  
    stampaVett(a);  
}
```

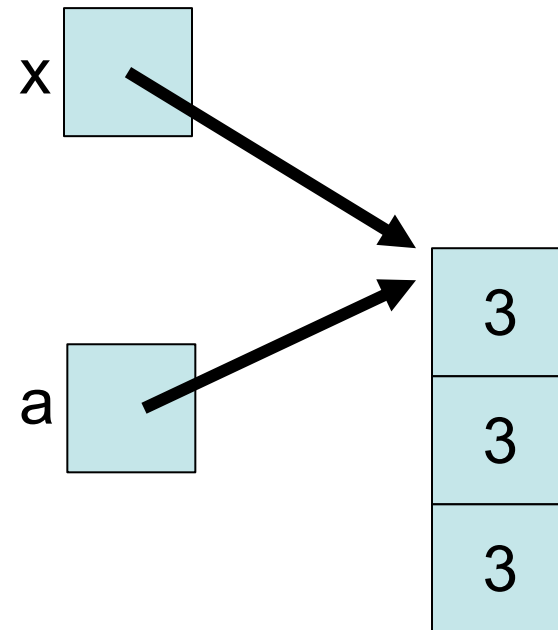


- valuto parametro attuale (valore: riferimento all'array {3, 3, 3})
- creo parametro formale (riferimento x)
- inizializzo il parametro formale (x ora è un riferimento all'array {3, 3, 3})

Passaggio di riferimenti

```
public static void incrementa(int[] x) {  
    for (int i=0; i<x.length; i++)  
        x[i]++;  
}
```

```
public static void main(String[] s) {  
    int[] a = {3, 3, 3};  
    stampaVett(a);  
    incrementa(a);  
    stampaVett(a);  
}
```

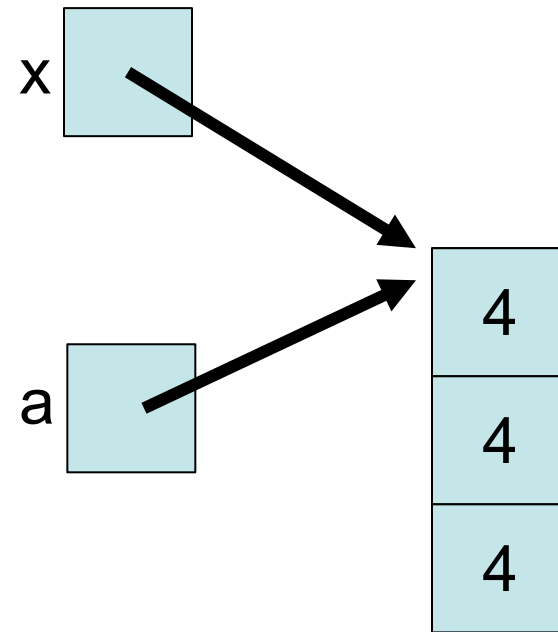


- valuto parametro attuale (valore: riferimento all'array {3, 3, 3})
- creo parametro formale (riferimento x)
- inizializzo il parametro formale (x ora è un riferimento all'array {3, 3, 3})
- eseguo il metodo (incrementa gli elementi dell'array, portandoli a {4, 4, 4})

Passaggio di riferimenti

```
public static void incrementa(int[] x) {  
    for (int i=0; i<x.length; i++)  
        x[i]++;  
}
```

```
public static void main(String[] s) {  
    int[] a = {3, 3, 3};  
    stampaVett(a);  
    incrementa(a);  
    stampaVett(a);  
}
```

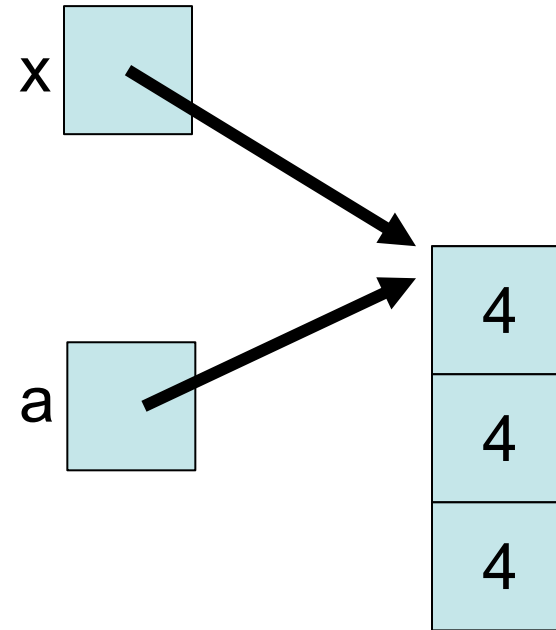


- valuto parametro attuale (valore: riferimento all'array {3, 3, 3})
- creo parametro formale (riferimento x)
- inizializzo il parametro formale (x ora è un riferimento all'array {3, 3, 3})
- eseguo il metodo (incrementa gli elementi dell'array, portandoli a {4, 4, 4})

Passaggio di riferimenti

```
public static void incrementa(int[] x) {  
    for (int i=0; i<x.length; i++)  
        x[i]++;  
}
```

```
public static void main(String[] s) {  
    int[] a = {3, 3, 3};  
    stampaVett(a);  
    incrementa(a);  
    stampaVett(a);  
}
```

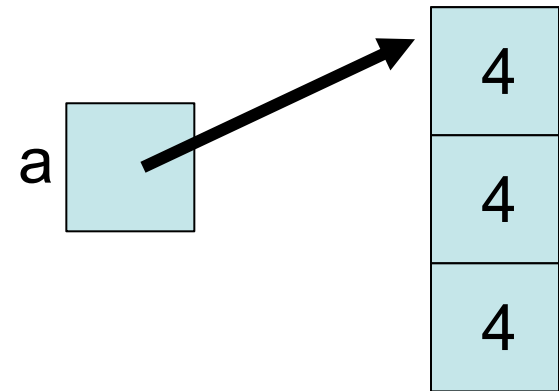


- valuto parametro attuale (valore: riferimento all'array {3, 3, 3})
- creo parametro formale (riferimento x)
- inizializzo il parametro formale (x ora è un riferimento all'array {3, 3, 3})
- eseguo il metodo (incrementa gli elementi dell'array, portandoli a {4, 4, 4})
- al termine del metodo le variabili locali vengono eliminate

Passaggio di riferimenti

```
public static void incrementa(int[] x) {  
    for (int i=0; i<x.length; i++)  
        x[i]++;  
}
```

```
public static void main(String[] s) {  
    int[] a = {3, 3, 3};  
    stampaVett(a);  
    incrementa(a);  
    stampaVett(a);  
}
```



- valuto parametro attuale (valore: riferimento all'array {3, 3, 3})
- creo parametro formale (riferimento x)
- inizializzo il parametro formale (x ora è un riferimento all'array {3, 3, 3})
- eseguo il metodo (incrementa gli elementi dell'array, portandoli a {4, 4, 4})
- al termine del metodo le variabili locali vengono eliminate