



Parameter Passing





What is Parameter Passing?

Parameters are:

- the **variables** or **arrays** that are passed in or out of procedures.
- the **variables** or **arrays** that are passed into functions



What is Parameter Passing?

The program below uses three procedures.

Main Program

```
SET UserLen TO 0  
SET UserBre TO 0  
SET TotalArea TO 0
```

```
CALL GetDimensions (UserLen, UserBre)  
CALL CalculateArea (UserLen, UserBre, Area)  
CALL DisplayArea (TotalArea)
```

The variables in brackets are **parameters** required by the procedures

These parameters are sent to the procedures to be used.

Each parameter is known as an **argument**.



What is Parameter Passing?

Procedures must be **declared** before they are **called**.

Main Program

```
...  
CALL GetDimensions (UserLen, UserBre)  
...
```

This line **CALLS** the procedure

GetDimensions

```
PROCEDURE GetDimensions (length, breadth)  
  
  RECEIVE length FROM KEYBOARD  
  RECEIVE breadth FROM KEYBOARD  
  
END PROCEDURE
```

This section **DECLARES** the procedure



What is Parameter Passing?

The **order** in which parameters are listed is important.

Main Program

...
CALL **GetDimensions** (*UserLen*, *UserBre*)
...

Notice the **order**
of the parameters
here...

GetDimensions

PROCEDURE **GetDimensions** (*length*, *breadth*)

RECEIVE *length* FROM KEYBOARD
RECEIVE *breadth* FROM KEYBOARD

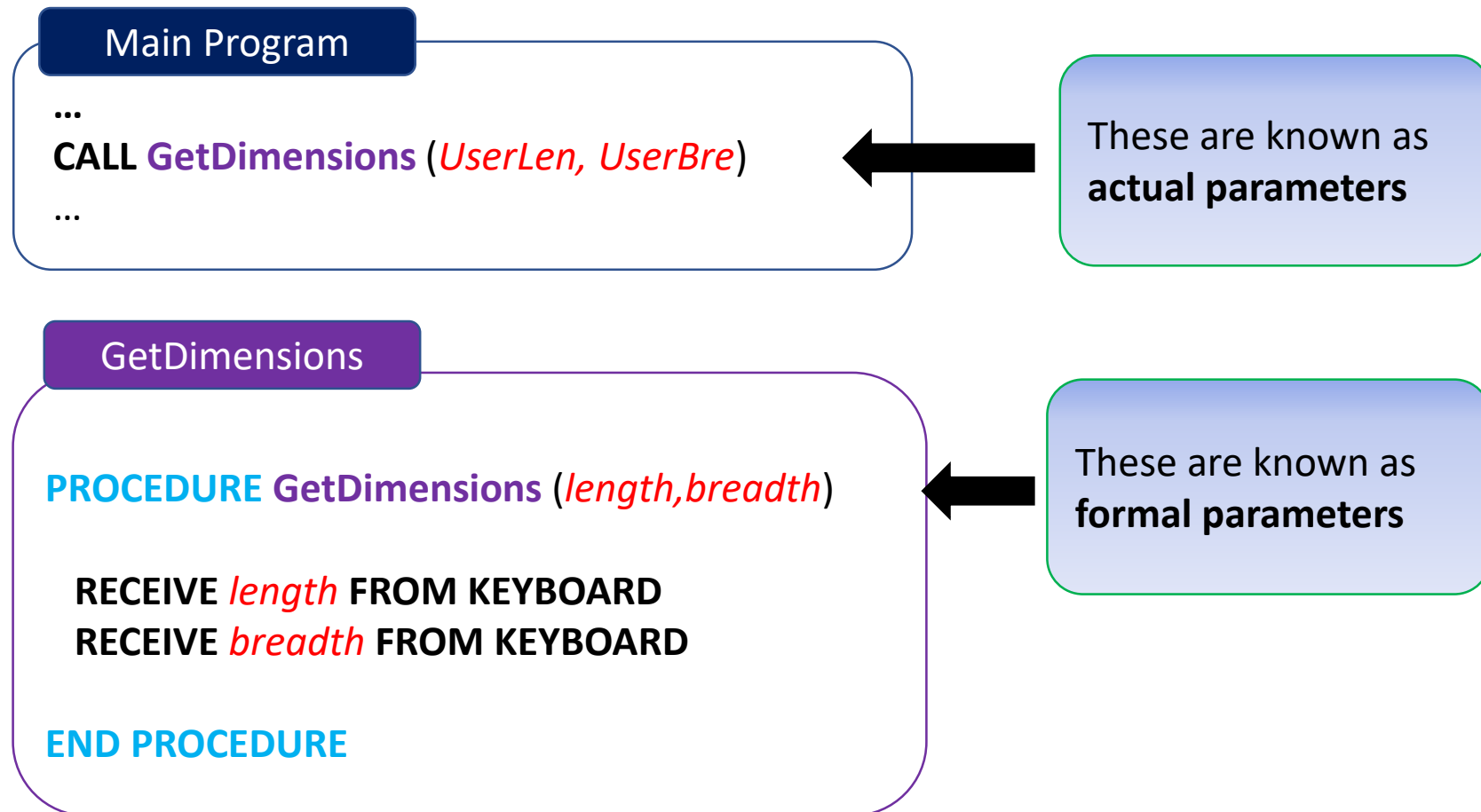
END PROCEDURE

...must be the same
as the order here –
but the **names** can
be different



What is Parameter Passing?

Parameters can be **actual** or **formal**.





What is Parameter Passing?

Actual parameters contain the value which is to be passed to the sub-program's formal parameter.

Formal parameters are used by the sub-program and contain a copy of the values passed from the actual parameters.

What is Parameter Passing?



Main Program

```
CALL GetDimensions (UserLen, UserBre)  
CALL CalculateArea (UserLen, UserBre, Area)  
CALL DisplayArea (TotalArea)
```

GetDimensions

```
PROCEDURE GetDimensions (length, breadth)
```

```
RECEIVE length FROM KEYBOARD  
RECEIVE breadth FROM KEYBOARD
```

```
END PROCEDURE
```

DisplayArea

```
PROCEDURE DisplayArea (Area)
```

```
SEND Area TO DISPLAY
```

```
END PROCEDURE
```

CalculateArea

```
PROCEDURE CalculateArea (length, breadth, Area)
```

```
SET Area TO length * breadth
```

```
END PROCEDURE
```




How are parameters passed?

When procedures are declared, the parameters they use can be set up to be passed in one of two ways.

- **By Value**
- **By Reference**



Pass by Reference or Value?

Any parameter whose **value is changed** by a procedure must be passed **by reference**

GetDimensions

PROCEDURE GetDimensions (*length*, *breadth*)

RECEIVE *length* FROM KEYBOARD

RECEIVE *breadth* FROM KEYBOARD

END PROCEDURE

How would length and breadth be passed to this procedure?



Pass by Reference or Value?

Any parameter whose **value is changed** by a procedure must be passed **by reference**

GetDimensions

PROCEDURE GetDimensions (Ref *length*, Ref *breadth*)

RECEIVE *length* FROM KEYBOARD

RECEIVE *breadth* FROM KEYBOARD

END PROCEDURE

length and breadth
will pass by
Reference

The values of **length** and
breadth are being used
to store new values
entered by the user

So the values of
length and **breadth**
are being **CHANGED**



Passing by Reference

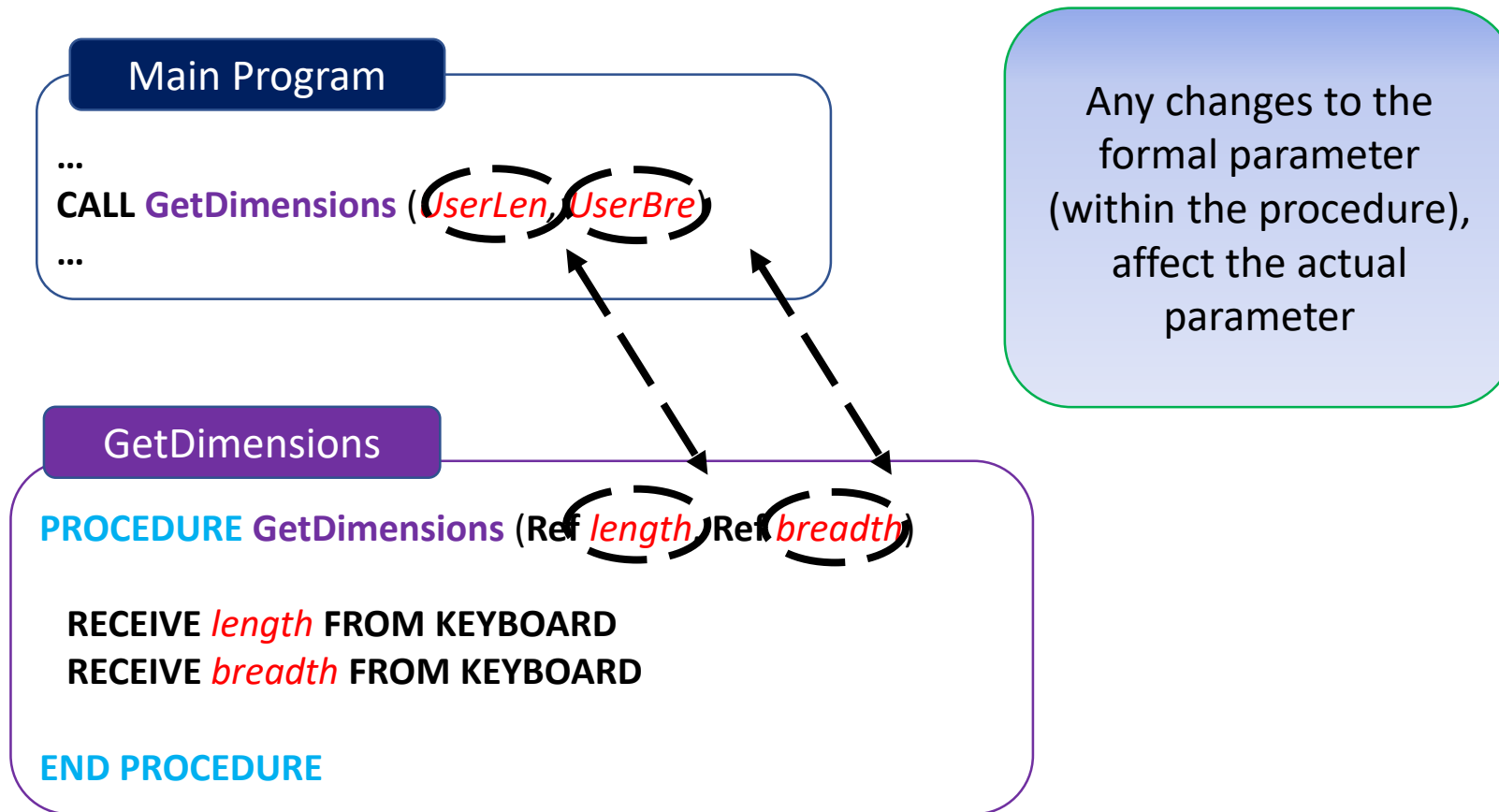
When a parameter is passed by reference, a link between the formal and actual parameters is created.

Any changes made to the formal parameter will also change the actual parameter.



Passing by Reference

A link between the formal and actual parameters is created





Pass by Reference or Value?

Any parameter whose **value is not changed** by a procedure should be passed **by value**

CalculateArea

```
PROCEDURE CalculateArea (length, breadth, Area)
```

```
SET Area TO length * breadth
```

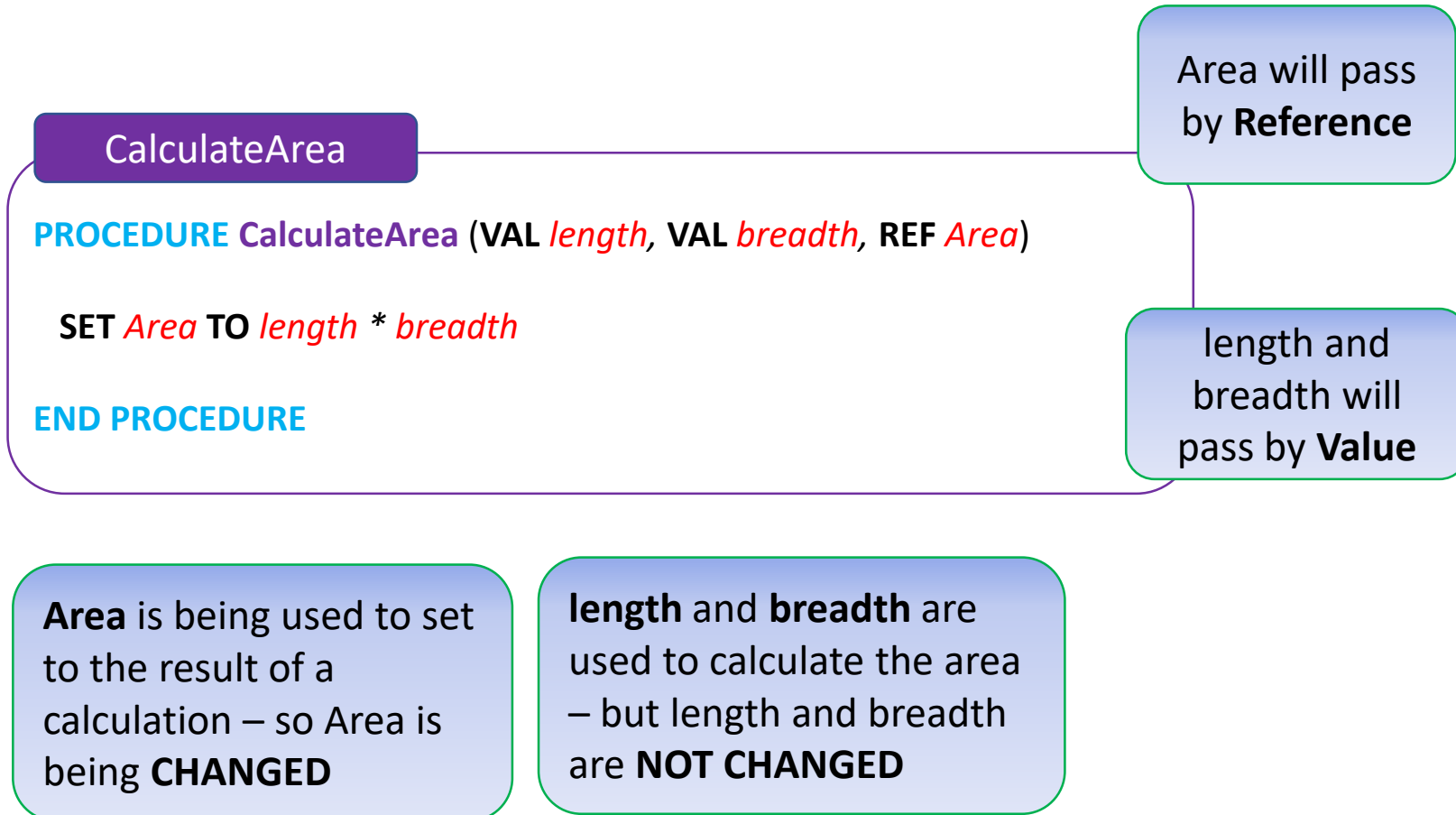
```
END PROCEDURE
```

How would length, breadth and area be passed to this procedure?



Pass by Reference or Value?

Any parameter whose **value** is not changed by a procedure should be passed **by value**



Passing by Value



When a parameter is passed by value, a copy of the actual parameter is sent to the procedure.

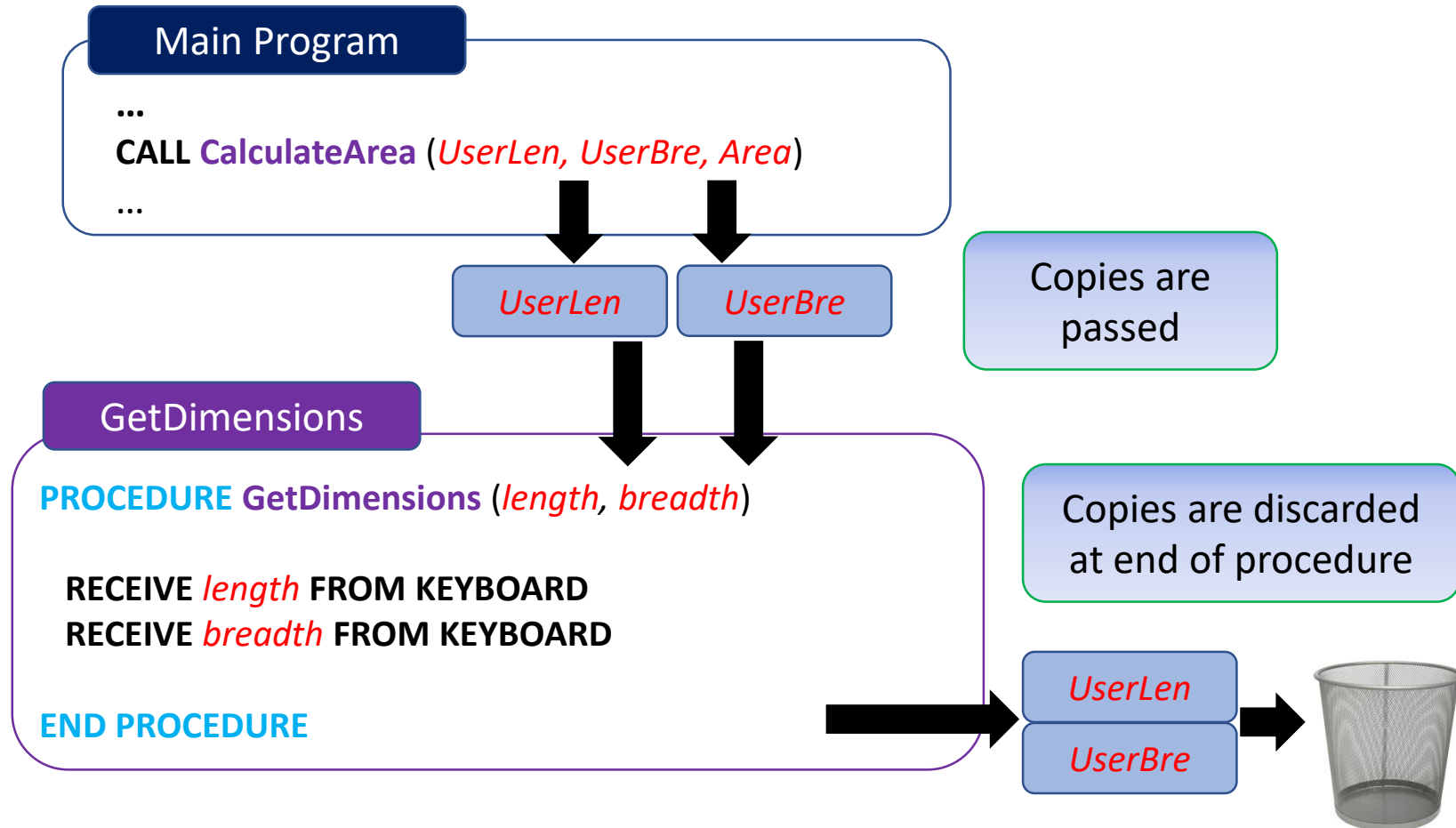
The formal parameter uses the copy and any changes it makes do not affect the actual parameter.

When the procedure is complete, the copy is discarded.



Passing by Value

A **copy** of the actual parameter is sent to the procedure.





Pass by Reference or Value?

Parameters used in several procedures will be passed in different ways depending on how the procedure uses them

DisplayArea

```
PROCEDURE DisplayArea (Area)
```

```
  SEND Area TO DISPLAY
```

```
END PROCEDURE
```

How would Area be passed to this procedure?



Pass by Reference or Value?

Parameters used in several procedures will be passed in different ways depending on how the procedure uses them

DisplayArea

```
PROCEDURE DisplayArea (VAL Area)
```

```
    SEND Area TO DISPLAY
```

```
END PROCEDURE
```

Area will pass
by **Value**

Area is being used
to display its value
on screen

But **Area** is
NOT CHANGED