Call by Reference

23-11-25

Aim – Write two programs to demonstrate call by value and call by reference respectively

Theory –
Call by value passes copies of variables to functions, so original data remains unchanged.
Call by reference passes addresses, allowing functions to modify actual variables directly in memory.

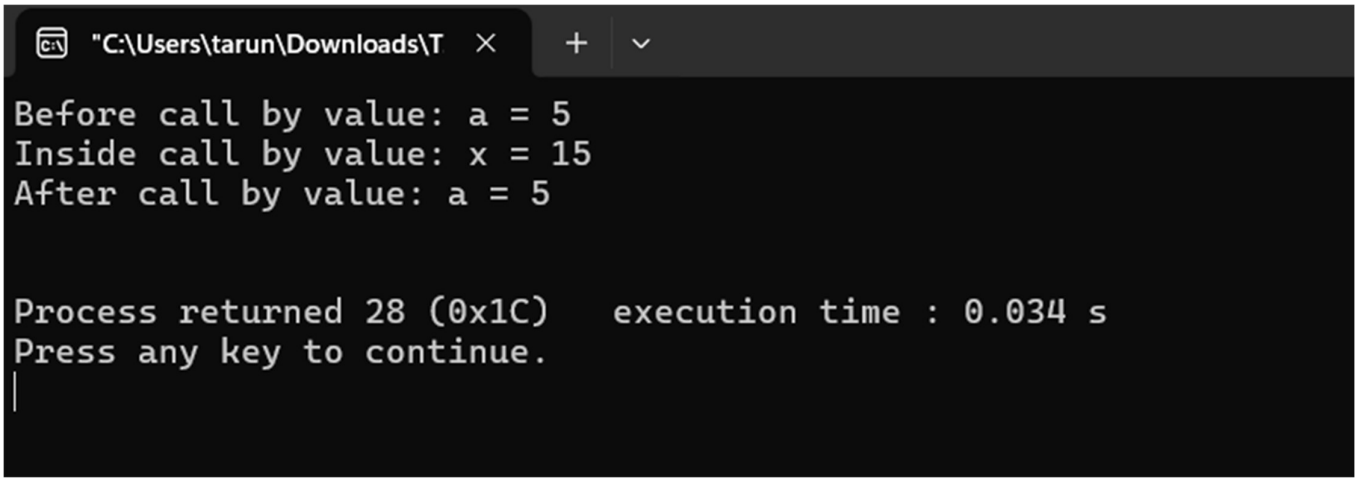
A1.

```
#include <stdio.h>


void cbv(int x) {
    x = x + 10;
    printf("Inside call by value: x = %d\n", x);
}


void main() {
    int a = 5;


    printf("Before call by value: a = %d\n", a);
    cbv(a);
    printf("After call by value: a = %d\n\n", a);
}
```

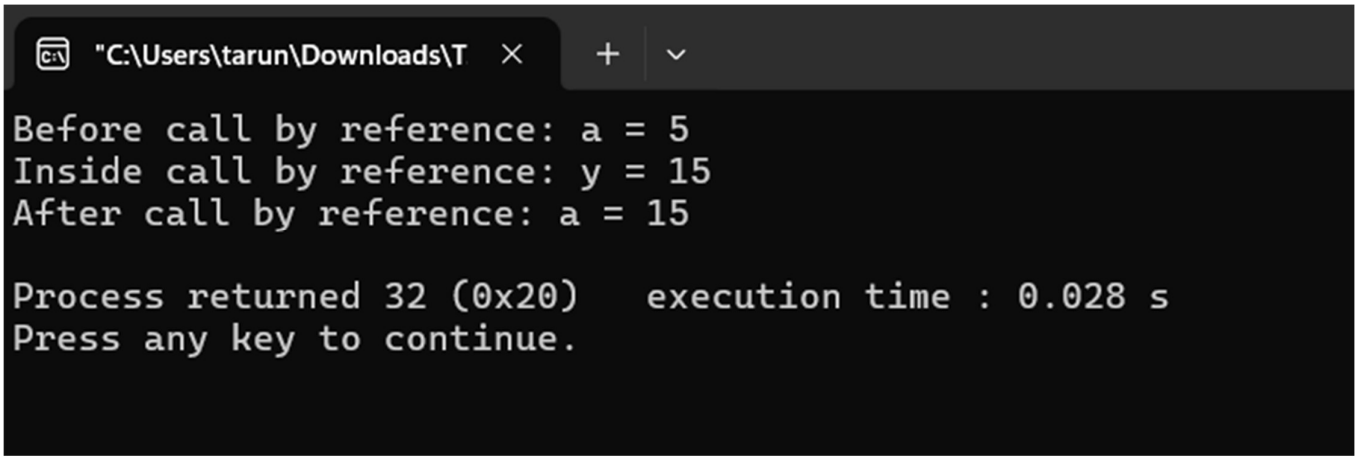


A2.

```
#include <stdio.h>
void cbr(int *y) {
    *y = *y + 10;
```

```
    printf("Inside call by reference: y = %d\n", *y);
}


void main() {
    int a = 5;
    printf("Before call by reference: a = %d\n", a);
    cbr(&a);
    printf("After call by reference: a = %d\n", a);
}
```



Conclusion

This program uses call by value and call by reference in order to not change and change the values in the original memory location inside the functions.