

### Task #03:

```
int a = 10;  
int *ptr = &a;
```

```
cout << *ptr << endl; // deref. // output = 10.  
&a; // 0x1661f6068.
```

ptr; // same address 0x1661f6068

\*ptr; // 10

\*ptr = 7;

~~a~~; // output = 7.  
&ptr; // 0x1661f6068

\*&a; // 7.

a = 2;

\*&\*&a; // 2

&\*ptr; // ptr address.

\* &\*ptr;

## Task 4:

arr [5] = { 1, 21, 3, 4, 5 };

<< arr << endl; // 0x16d117050.  
<< \*arr << endl; // 1  
<< (arr+2) << endl; // 0x16d117058  
<< \*(arr+2); // 3.  
<< \*arr - 4 // 1-4 = -3.

\*ptr = arr;

## Task 5:

int a = 10;  
int b = 5;  
const int \*ptr;

ptr = &a;  
ptr = &b;  
cout << \*ptr; // 5.

## Task 6:

const int a = 10;  
int b = 7;  
const int \*ptr;  
ptr = &a;

$*ptr = 20;$  // value.

`cout << *ptr ;`

### Task #08:

$a = 10;$

$b = 5;$

`int *const ptr = &a;`

`cout << ptr;` // address of  $a$ .

`cout << *ptr;` // value of  $a$  i.e 10.

$*ptr = 66 ;$

`cout << a++ << endl;` // 66

`cout << a << endl` // 67.