

Tutorial 11 Pointers and Call by Reference

1. Download ex1.cpp and ex2.cpp Compile and execute the program. Explain the output.

<i>File</i>	<i>Program segment</i>	<i>Output</i>
Ex1.c	<pre>#include <iostream> using namespace std; void main(){ int v=5, *ptr; ptr = &v; *ptr = 42; cout << "v=" << v << endl; cin>>*ptr; //Let's enter 100 //What happens if you write cin>>ptr; ? cout << "v=" << v << endl; v=7; cout << "*ptr is " << *ptr <<endl; cout<<"Address of v is"<<ptr << endl; }</pre>	
Ex2.c	<pre>#include <iostream> using namespace std; void f(int *a, int *b){ int *c; c=a; *c=*c+10; *b=*b+10; } void main(){ int x=3, y=4; int *ptr1; ptr1=&x; f(ptr1, &y); cout<< "x=" << x << endl; cout<< "y=" << y << endl; cout<<"*ptr1=" << *ptr1 << endl; }</pre>	

2) Download `current.cpp`

Modify the program to compute the value of current (I), given the values of resistance (R) and voltage (V), where

$$I = V/R$$

For example, when voltage=5.1 and resistance=2, the current is $5.1/2=2.55$

The program itself is easy, but for the sake of practice, we'd like to create three functions as follows:

`getInput`: The return type of this function is `void`. The function internally gets the resistance and voltage with `cin`. The variables R and V are then sent back to caller using `pass-by-pointer`.

`toCurrent`: calculate the current I given R and V . The function should return a real number for the value of I . Although simple pass-by-value also works, in here, let's try to pass R and V using `pass-by-reference`.

`main`: calls `getInput()` to obtain V and R by using `pass-by-pointer`. After that, the program should pass V and R to the function `toCurrent()` to obtain I . Finally, the value of I is printed (to 2 decimal places, with `cout << fixed << setprecision(2)`)

Sample input/output:

Please enter the resistant: <u>5</u>
Please enter the voltage: <u>2</u>
The value of current is 0.40