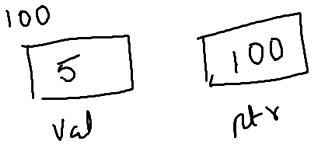
Pointers in Functions
Double Pointers
Practice Problems

Part - 2.

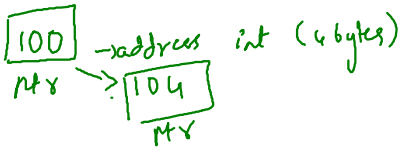
-> Pointers in functions.

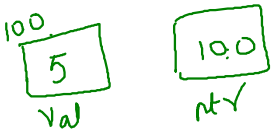
```
print(int ptr) {        // ptr = 100
  cout << ptr.  // 100
}

main() {
  int val = 5;
  int* ptr = &val;
  val      // 5
  *ptr     // 5
  print(ptr)
}
```

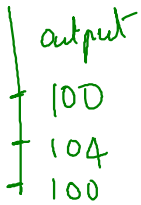


① Pass by value ✓
② Pass by reference.
③ Passing by pointer

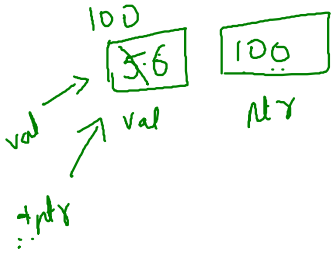
```
void print(int *ptr){          // ptr = 5
    ptr=ptr+1;
    cout<<"Inside print: "<<ptr<<endl;
}
int main(){
    int val=5;
    int* ptr=&val;
    cout<<"Before: "<<ptr<<endl;
    print(ptr);
    cout<<"After: "<<ptr<<endl;
}
```

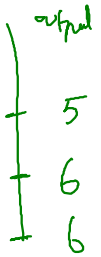







```
void print(int *ptr){          // ptr = 5
    *ptr=*ptr+1;               // 5+1 => 6
    cout<<"Inside print: "<<*ptr<<endl;
}
int main(){
    int val=5;
    int* ptr=&val;
    cout<<"Before: "<<*ptr<<endl;
    print(ptr);
    cout<<"After: "<<*ptr<<endl;
}
```



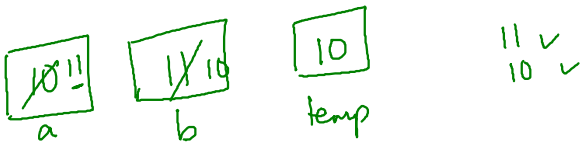


Swap 2 numbers

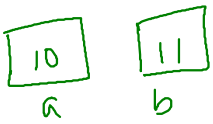
a=10 , b=11

a=11 b=10

```
void swap(int a,int b){
    int temp=a;
    a=b;
    b=temp;
    cout<<"A "<<a<<endl;
    cout<<"B "<<b<<endl;
}
int main(){
    int a=10,b=11;
    swap(a,b);
    cout<<"Main function"<<endl;
    cout<<"A "<<a<<endl;
    cout<<"B "<<b<<endl;
}
```



11
10



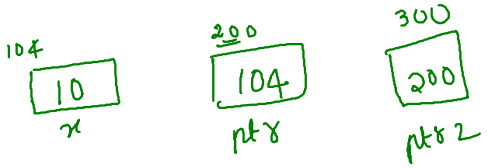
10
11

```
void swap(int* a,int* b){
    int temp=*a;
    *a=*b;
    *b=temp;
    cout<<"A "<<*a<<endl;
    cout<<"B "<<*b<<endl;
}
int main(){
    int a=10,b=11;
    int* ptr1=&a;
    int* ptr2=&b;
    swap(ptr1,ptr2);
    cout<<"Main function"<<endl;
    cout<<"A "<<a<<endl;
    cout<<"B "<<b<<endl;
}
```

11
10
11
10

→ Double pointers.

```
int x = 10;
int * ptr = &x;
int ** ptr2 = &ptr
```



x → 104

ptr → 200

ptr2 → 300

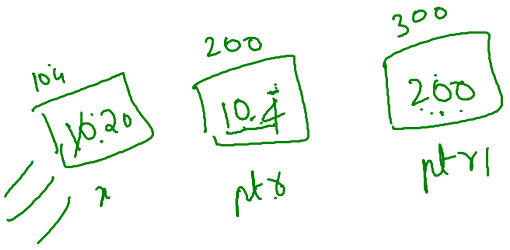
```
int main(){
    int x=10;
    int* ptr=&x;
    int** ptr1=&ptr;
    cout<<x<<endl;
    cout<<&x<<endl;
     cout<<ptr<<endl;
    cout<<&ptr<<endl;
      cout<<ptr1<<endl;
    cout<<&ptr1<<endl;
}
```

10

+ptr => add

++ptr => poin

```
    int x=10;
    int* ptr=&x;
    int** ptr1=&ptr;
    cout<<x<<endl;
    cout<<*ptr<<endl;
    cout<<**ptr1<<endl;
```



**ptr1 = ++ptr1 + 10;

10+10

x => 20

*ptr => 20

*ptr1 => 20

*ptr -> 1level

**ptr -> 2level

***ptr => 3level.

Solving Problems

16 -> Pointers

1

```
void problem1() {
    int alpha = 7;
    int beta = 15;
    int *ptr = &beta;
    *ptr = 10;
    cout << alpha << " " << beta << endl;
}
int main() {
    problem1();
    return 0;
}
```

7 10

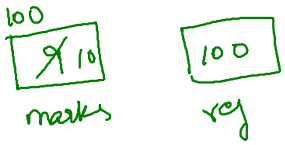
2

```
void problem2() {
    int score = 5;
    int *a = &score;
    int *b = a;
    (*b)++;
    cout << score << endl;

}
int main() {
    problem2();
    return 0;
}
```

6

3

```
void problem3() {
    int marks = 9;
    int *ref = &marks;
    cout << (*ref)++ << " ";
    cout << marks << endl;
}
int main() {
    problem3();
    return 0;
}
```

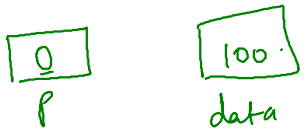


9
10

4

```
void problem4() {
    int *p = 0;
    int data = 100;
    *p = data;
    cout << *p;

}

int main() {
    problem4();
    return 0;
}
```

not pointing anywhere
segmentation fault



5

```
void problem5() {
    int x = 3;
    int y = 5;
    int *z = &y;
    x = *z;
    *z = *z + 3;
    cout << x << "  " << y << endl;
}
int main() {
    problem5();
    return 0;
}
```

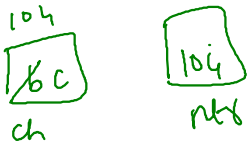
25.5
25.5
25.5

6

```
void problem6() {
    float x = 15.5;
    float y = 25.5;
    float *ptr = &x;
    (*ptr)++;
    *ptr = y;
    cout << *ptr << " " << x << " " << y << endl;
}
int main() {
    problem6();
    return 0;
}
```

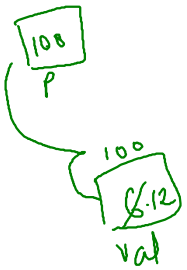
c

```
void problem7() {
    char ch = 'b';
    char* ptr = &ch;
    ch++;
    cout << *ptr << endl;
}
int main() {
    problem7();
    return 0;
}
```



8

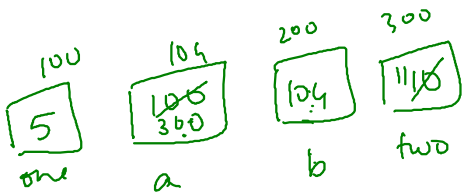
```
void multiply(int *p){
    *p = (*p)  * 2;
}
void problem8() {
    int val = 6;
    multiply(&val);
    cout << val << endl;
}
int main() {
    problem8();
    return 0;
}
```



9

```
void problem9() {
    int one = 5;
    int *a = &one;
    int **b = &a;
    int two = 10;
    *b = &two;
    (*a)++;
    cout << one << " " << two << endl;
}
int main() {
    problem9();
    return 0;
}
```

5
11

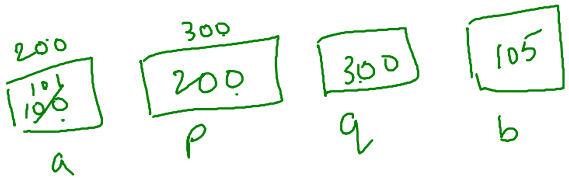


10

```
void problem10() {
    int a = 100;
    int *p = &a;
    int **q = &p;
    int b = (**q)++ + 5;
    cout << a << " " << b << endl;
}
int main() {
    problem10();
    return 0;
}
```

101 105

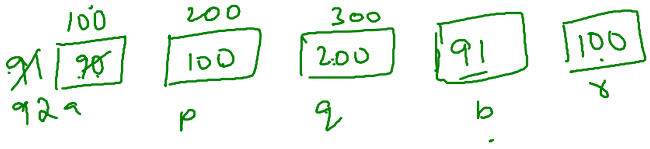
100+5 => 105



11

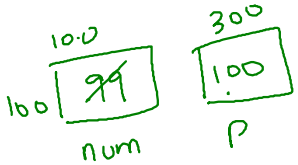
```
void problem11() {
    int a = 90;
    int *p = &a;
    int **q = &p;
    int b = ++(**q);
    int *r = *q;
    ++(*r);
    cout << a << " " << b << endl;
}
int main() {
    problem11();
    return 0;
}
```

92
91



12

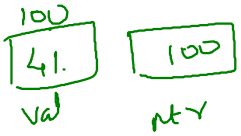
```
void increment(int **ptr){
    ++(**ptr);
}
void problem12() {
    int num = 99;
    int *p = &num;
    increment(&p);
    cout << num << endl;
}
int main() {
    problem12();
    return 0;
}
```



100

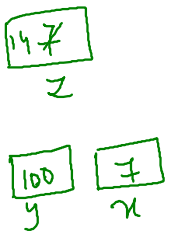
(13)

```
void problem13() {
    int val = 41;
    int *ptr = &val;
    cout << *ptr + 8 << endl;
}
int main() {
    problem13();
    return 0;
}
```

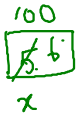


41+8 => 9

```
#include <iostream>
using namespace std;

void R(int z) {
    z += z;
    cout<<z << " ";
}
void S(int *y)  {
    int x = *y + 2;
    R(x);
    *y = x - 1;
    cout<<x << " ";
}
void problem14() {
    int x = 5;
    S(&x);
    cout<<x << endl;
}
int main() {
    problem14();
    return 0;
}
```

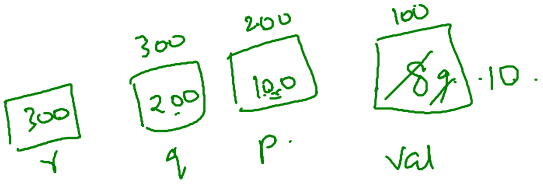


14
7
6



(15)

```
#include <iostream>
using namespace std;

void problem15() {
    int ***r, **q, *p, val = 8;
    p = &val;
    (*p)++;
    q = &p;
    (**q)++;
    r = &q;
    cout << *p << " " << **q << " " << ***r << endl;
}
int main() {
    problem15();
    return 0;
}
```



10
10
10

```
#include <iostream>
using namespace std;

int compute(int x, int *y, int **z) {
    int a, b;
    **z += 1;
    b = **z;
    *y += 2;
    a = *y;
    x += 3;
    return x + a + b;
}
void problem16() {
    int c, *b, **a;
    c = 4;
    b = &c;
    a = &b;
    cout << compute(c, b, a) << endl;
}
int main() {
    problem16();
    return 0;
}
```

Homework

```
#include <iostream>
using namespace std;

int main() {
    int x = 10;
    int *p = &x;
    cout << p << endl;
    p = p + 1;
    cout << p << endl;
    return 0;
}
```

```
#include <iostream>
using namespace std;

void updateValue(int *ptr) {
    *ptr = *ptr * 2 + 1;
}

int main() {
    int x = 7;
    updateValue(&x);
    cout << x << endl;
    return 0;
}
```

```
#include <iostream>
using namespace std;

int main() {
    int a = 100;
    int *p = &a;
    int **q = &p;

    cout << **q << endl;
    **q = **q + 50;
    cout << a << endl;

    return 0;
}
```

```
#include <iostream>
using namespace std;

int main() {
    int a = 5;
    int b = 15;
    int *p = &a;
    *p = b;
    p = &b;
    *p = 30;

    cout << a << " " << b << endl;
    return 0;
}
```

```
#include <iostream>
using namespace std;

int main() {
    int x = 9;
    const int *ptr = &x;
    x = x + 1;
    cout << *ptr << endl;
    return 0;
}
```