## **Pointers Revisited**

Quan Thanh Tho

```
int *p;
cout << p << * p;</pre>
```

```
int *p = new int;
cout << p << * p;
```

```
int *p = new int;
p = 100;
cout << p << * p;</pre>
```

```
int *p = new int;
*p = 100;
cout << p << * p;</pre>
```

```
int *p = new int; int * p2;
*p = 100;
*p2 = *p;
cout << *p << *p2;</pre>
```

```
int *p = new int; int * p2 = new int;
*p = 100;
*p2 = *p;
cout << *p << *p2;</pre>
```

```
int *p = new int; int * p2 = new int;
*p = 100;
p2 = p;
cout << *p << *p2;</pre>
```

```
int *p = new int;
int* p2 = new int;
int* p3 = new int
*p = 100;
p2 = p; *p3 = *p2;
cout << *p << *p2<<*p3;</pre>
```

```
int *p = new int;
int* p2 = new int;
int* p3 = new int
*p = 100;
p2 = p; *p3 = *p2;
*p = 4;
cout << *p << *p2<<*p3;
```

```
int *p = new int;
                           C *p = new C();
int* p2 = new int;
                           C^* p2 = new C();
int* p3 = new int
                           C^* p3 = new C();
*p = 100;
                           p->x = 100;
p2 = p; *p3 = *p2;
                           p2 = p; p3->x= p2->x;
*p = 4;
                            p->x=4;
cout << *p << *p2 << *p3; ...p->x << p2->x << p3->x;
```

```
C *p = new C();
C p = new C();
C p2 = new C();
                            C^* p2 = new C();
C p3 = new C();
                            C^* p3 = new C();
p.x = 100;
                            p->x = 100;
p2 = p; p3 .x = p2.x;
                            p2 = p; p3->x= p2->x;
p.x = 4;
                            p->x=4;
...p.x << p2.x << p3.x...;
                            ...p->x << p2->x<<p3->x;
```