Discussion Week 1

Tuesday

October 6, 2020

Sigal Shaul

About Discussions

- Scheduled for 1 hour
  - 30 minutes content/practice, 30 minutes Q&A
- Focusing on PA or Content Review
- Friday Discussions
  - Introduction to PA
  - Review of the week's material
- Tuesday Discussions
  - PA common problems/Q&A
  - Review of the week's material

Today

C++ Review

What will be printed when this code executes?

```
void mystery(int* ptr, int y) {
     y = 4;
     ptr = &y;
}


int main() {
     int x = 15;
     int y = 21;
     int* ptr = &x;
     mystery(ptr, y);
     cout << *ptr;
     return 0;
}
```

a) 15
b) 21
c) 4
d) Error
e) Something else

What will be printed when this code executes?

```
void mystery(int* ptr, int y) {
     y = 4;
     ptr = &y;
}


int main() {
     int x = 15;
     int y = 21;
     int* ptr = &x;
     mystery(ptr, y);
     cout << *ptr;
     return 0;
}
```

a) **15**
b) 21
c) 4
d) Error
e) Something else

What will be printed when this code executes?

```
void mystery(int* ptr, int y) {
    y = 4;
    ptr = &y;
}
```

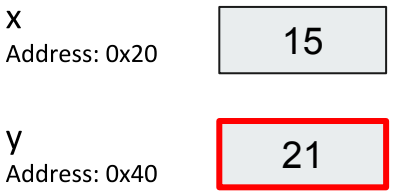
```
int main() {
    int x = 15;
    int y = 21;
    int* ptr = &x;
    mystery(ptr, y);
    cout << *ptr;
    return 0;
}
```

x
Address: 0x40

What will be printed when this code executes?

```
void mystery(int* ptr, int y) {
    y = 4;
    ptr = &y;
}
```

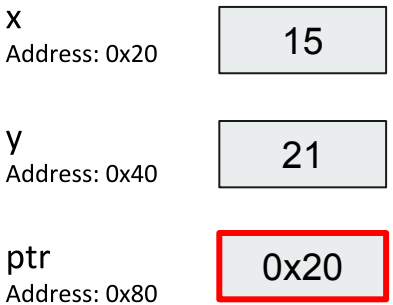
```
int main() {
    int x = 15;
    int y = 21;
    int* ptr = &x;
    mystery(ptr, y);
    cout << *ptr;
    return 0;
}
```



What will be printed when this code executes?

```
void mystery(int* ptr, int y) {
    y = 4;
    ptr = &y;
}
```

```
int main() {
    int x = 15;
    int y = 21;
    int* ptr = &x;
    mystery(ptr, y);
    cout << *ptr;
    return 0;
}
```



What will be printed when this code executes?

```
void mystery(int* ptr, int y) {
    y = 4;
    ptr = &y;
}
```

ptr
Address: 0x120
0x20

y
Address: 0x160
21

```
int main() {
    int x = 15;
    int y = 21;
    int* ptr = &x;
    mystery(ptr, y);
    cout << *ptr;
    return 0;
}
```

x
Address: 0x20
15

y
Address: 0x40
21

ptr
Address: 0x80
0x20

What will be printed when this code executes?

```
void mystery(int* ptr, int y) {
    y = 4;
    ptr = &y;
}
```

ptr
Address: 0x120
0x20

y
Address: 0x160
4

```
int main() {
    int x = 15;
    int y = 21;
    int* ptr = &x;
    mystery(ptr, y);
    cout << *ptr;
    return 0;
}
```

x
Address: 0x20
15

y
Address: 0x40
21

ptr
Address: 0x80
0x20

What will be printed when this code executes?

```
void mystery(int* ptr, int y) {
    y = 4;
    ptr = &y;
}
```

ptr Address: 0x120	~~0x20~~ 0x160
y Address: 0x160	4

```
int main() {
    int x = 15;
    int y = 21;
    int* ptr = &x;
    mystery(ptr, y);
    cout << *ptr;
    return 0;
}
```

x Address: 0x20	15
y Address: 0x40	21
ptr Address: 0x80	0x20

What will be printed when this code executes?

```
void mystery(int* ptr, int y) {
     y = 4;
     ptr = &y;
}
```

ptr
Address: 0x120

0x160

y
Address: 0x160

4

```
int main() {
     int x = 15;
     int y = 21;
     int* ptr = &x;
     mystery(ptr, y);
     cout << *ptr;
     return 0;
}
```

```
Output is:

15
```

x
Address: 0x20

15

y
Address: 0x40

21

ptr
Address: 0x80

0x20

What will be printed when this code executes?

```
void mystery(int* ptr, int y) {
     y = 4;
     *ptr = y;
}


int main() {
     int x = 15;
     int y = 21;
     int* ptr = &x;
     mystery(ptr, y);
     cout << *ptr;
     return 0;
}
```

a) 15
b) 21
c) 4
d) Error
e) Something else

What will be printed when this code executes?

```
void mystery(int* ptr, int y) {
     y = 4;
     *ptr = y;
}


int main() {
     int x = 15;
     int y = 21;
     int* ptr = &x;
     mystery(ptr, y);
     cout << *ptr;
     return 0;
}
```

a) 15
b) 21
c) 4
d) Error
e) Something else

What will be printed when this code executes?

```
void mystery(int* ptr, int y) {
    y = 4;
    *ptr = y;
}
```

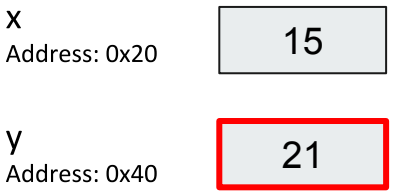
```
int main() {
    int x = 15;
    int y = 21;
    int* ptr = &x;
    mystery(ptr, y);
    cout << *ptr;
    return 0;
}
```

x
Address: 0x20

What will be printed when this code executes?

```
void mystery(int* ptr, int y) {
    y = 4;
    *ptr = y;
}
```

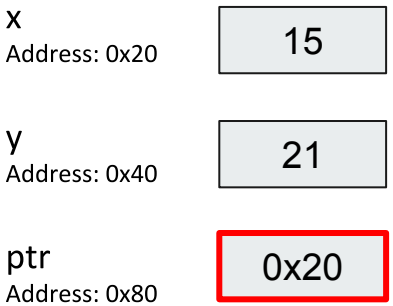
```
int main() {
    int x = 15;
    int y = 21;
    int* ptr = &x;
    mystery(ptr, y);
    cout << *ptr;
    return 0;
}
```



What will be printed when this code executes?

```
void mystery(int* ptr, int y) {
    y = 4;
    *ptr = y;
}
```

```
int main() {
    int x = 15;
    int y = 21;
    int* ptr = &x;
    mystery(ptr, y);
    cout << *ptr;
    return 0;
}
```



What will be printed when this code executes?

```
void mystery(int* ptr, int y) {
    y = 4;
    *ptr = y;
}
```

ptr
Address: 0x120
0x20

y
Address: 0x160
21

```
int main() {
    int x = 15;
    int y = 21;
    int* ptr = &x;
    mystery(ptr, y);
    cout << *ptr;
    return 0;
}
```

x
Address: 0x20
15

y
Address: 0x40
21

ptr
Address: 0x80
0x20

What will be printed when this code executes?

```
void mystery(int* ptr, int y) {
    y = 4;
    *ptr = y;
}
```

ptr
Address: 0x120
0x20

y
Address: 0x160
4

```
int main() {
    int x = 15;
    int y = 21;
    int* ptr = &x;
    mystery(ptr, y);
    cout << *ptr;
    return 0;
}
```

x
Address: 0x20
15

y
Address: 0x40
21

ptr
Address: 0x80
0x20

What will be printed when this code executes?

```
void mystery(int* ptr, int y) {
    y = 4;
    *ptr = y;
}
```

ptr Address: 0x120	0x20
y Address: 0x160	4

```
int main() {
    int x = 15;
    int y = 21;
    int* ptr = &x;
    mystery(ptr, y);
    cout << *ptr;
    return 0;
}
```

x Address: 0x20	~~15~~ 4
y Address: 0x40	21
ptr Address: 0x80	0x20

What will be printed when this code executes?

```
void mystery(int* ptr, int y) {
     y = 4;
     ptr* = y;
}
```

ptr
Address: 0x120

0x20

y
Address: 0x160

4

```
int main() {
     int x = 15;
     int y = 21;
     int* ptr = &x;
     mystery(ptr, y);
     cout << *ptr;
     return 0;
}
```

Output is:

4

x
Address: 0x20

4

y
Address: 0x40

21

ptr
Address: 0x80

0x20

What's the Difference?

```
void mystery(int* ptr, int y) {
    y = 4;
    ptr = &y;
}

int main() {
    int x = 15;
    int y = 21;
    int* ptr = &x;
    mystery(ptr, y);
    cout << *ptr;
    return 0;
}
```

```
void mystery(int* ptr, int y) {
    y = 4;
    *ptr = y;
}

int main() {
    int x = 15;
    int y = 21;
    int* ptr = &x;
    mystery(ptr, y);
    cout << *ptr;
    return 0;
}
```

What's the Difference?

```
void mystery(int* ptr, int y) {
    y = 4;
    ptr = &y;
}

int main() {
    int x = 15;
    int y = 21;
    int* ptr = &x;
    mystery(ptr, y);
    cout << *ptr;
    return 0;
}
```

In the first case we are updating the value stored in `ptr` (a memory address). In the second case we are dereferencing `ptr` and updating the value stored at the memory address that is stored in `ptr`.

```
void mystery(int* ptr, int y) {
    y = 4;
    *ptr = y;
}

int main() {
    int x = 15;
    int y = 21;
    int* ptr = &x;
    mystery(ptr, y);
    cout << *ptr;
    return 0;
}
```

How can we modify this code?

```
void mystery(int* ptr2, int & y2) {
     y2 = 4;
     ptr2 = &y2;
}


int main() {
     int x = 15;
     int y = 21;
     int* ptr = &x;
     mystery(ptr, y);
     cout << *ptr;
     return 0;
}
```

Quick check: what does main print?

How can we modify this code?

```
void mystery(int* ptr2, int & y2) {
     y2 = 4;
     ptr2 = &y2;
}


int main() {
     int x = 15;
     int y = 21;
     int* ptr = &x;
     mystery(ptr, y);
     cout << *ptr;
     return 0;
}
```

Quick check: what does main print? 15

How can we modify this code?

```
void mystery(int* ptr2, int & y2) {
     y2 = 4;
     ptr2 = &y2;
}


int main() {
     int x = 15;
     int y = 21;
     int* ptr = &x;
     mystery(ptr, y);
     cout << *ptr;
     return 0;
}
```

Quick check: Draw memory model

What will be printed when this code executes?

```
void mystery(int* ptr2, int & y2)
{
     y2 = 4;
     ptr2 = &y2;
}
```

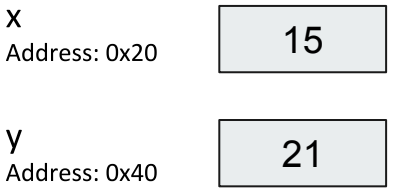
```
int main() {
     int x = 15;
     int y = 21;
     int* ptr = &x;
     mystery(ptr, y);
     cout << *ptr;
     return 0;
}
```

x
Address: 0x20

What will be printed when this code executes?

```
void mystery(int* ptr2, int & y2)
{
    y2 = 4;
    ptr2 = &y2;
}
```

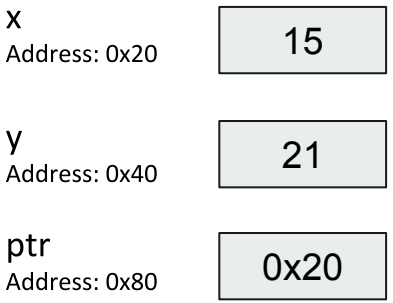
```
int main() {
    int x = 15;
    int y = 21;
    int* ptr = &x;
    mystery(ptr, y);
    cout << *ptr;
    return 0;
}
```



What will be printed when this code executes?

```
void mystery(int* ptr2, int & y2)
{
    y2 = 4;
    ptr2 = &y2;
}
```

```
int main() {
    int x = 15;
    int y = 21;
    int* ptr = &x;
    mystery(ptr, y);
    cout << *ptr;
    return 0;
}
```



What will be printed when this code executes?

```
void mystery(int* ptr2, int & y2)
{
     y2 = 4;
     ptr2 = &y2;
}
```

ptr2
Address: 0x120

0x20

```
int main() {
     int x = 15;
     int y = 21;
     int* ptr = &x;
     mystery(ptr, y);
     cout << *ptr;
     return 0;
}
```

x
Address: 0x20

15

y, y2
Address: 0x40

21

ptr
Address: 0x80

0x20

What will be printed when this code executes?

```
void mystery(int* ptr2, int & y2)
{
    y2 = 4;
    ptr2 = &y2;
}
```

ptr2
Address: 0x120

0x20

```
int main() {
    int x = 15;
    int y = 21;
    int* ptr = &x;
    mystery(ptr, y);
    cout << *ptr;
    return 0;
}
```

x
Address: 0x20

15

y, y2
Address: 0x40

4

ptr
Address: 0x80

0x20

What will be printed when this code executes?

```
void mystery(int* ptr2, int & y2)
{
     y2 = 4;
     ptr2 = &y2;
}
```

ptr2
Address: 0x120

0x40

```
int main() {
     int x = 15;
     int y = 21;
     int* ptr = &x;
     mystery(ptr, y);
     cout << *ptr;
     return 0;
}
```

x
Address: 0x20

15

y, y2
Address: 0x40

4

ptr
Address: 0x80

0x20

What will be printed when this code executes?

```
void mystery(int* ptr2, int & y2)
{
     y2 = 4;
     ptr2 = &y2;
}
```

ptr2
Address: 0x120

0x40

```
int main() {
     int x = 15;
     int y = 21;
     int* ptr = &x;
     mystery(ptr, y);
     cout << *ptr;
     return 0;
}
```

x
Address: 0x20

15

y, y2
Address: 0x40

4

ptr
Address: 0x80

0x20

What will be printed when this code executes?

```
void mystery(int* ptr2, int & y2)
{
     y2 = 4;
     ptr2 = &y2;
}
```

ptr2
Address: 0x120

0x40

```
int main() {
     int x = 15;
     int y = 21;
     int* ptr = &x;
     mystery(ptr, y);
     cout << *ptr;
     return 0;
}
```

```
Output is:

15
```

x
Address: 0x20

15

y, y2
Address: 0x40

4

ptr
Address: 0x80

0x20

How can we modify this code?

```
void mystery(int* ptr2, int & y2) {
     y2 = 4;
     ptr2 = &y2;
}


int main() {
     int x = 15;
     int y = 21;
     int* ptr = &x;
     mystery(ptr, y);
     cout << *ptr;
     return 0;
}
```

How can we change the code so this line modifies ptr in main?

How can we modify this code?

```
void mystery(int** ptr2, int & y2) {
     y2 = 4;
     *ptr2 = &y2;
}


int main() {
     int x = 15;
     int y = 21;
     int* ptr = &x;
     mystery(&ptr, y);
     cout << *ptr;
     return 0;
}
```

How can we change the code so this line modifies ptr in main?

One solution: Pass in a pointer to the pointer!

Let's look at the memory model

```
void mystery(int** ptr2, int & y2) {
    y2 = 4;
    *ptr2 = &y2;
}
```

ptr2
Address: 0x120

0x80

```
int main() {
    int x = 15;
    int y = 21;
    int* ptr = &x;
    mystery(&ptr, y);
    cout << *ptr;
    return 0;
}
```

x
Address: 0x20

15

y, y2
Address: 0x40

21

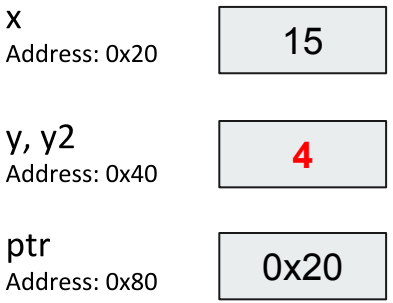
ptr
Address: 0x80

0x20

Let's look at the memory model

```
void mystery(int** ptr2, int & y2) {
     y2 = 4;
     *ptr2 = &y2;
}
```

```
int main() {
     int x = 15;
     int y = 21;
     int* ptr = &x;
     mystery(&ptr, y);
     cout << *ptr;
     return 0;
}
```



Let's look at the memory model

```
void mystery(int** ptr2, int & y2)
{
     y2 = 4;
     *ptr2 = &y2;
}
```

ptr2
Address: 0x120

0x80

```
int main() {
     int x = 15;
     int y = 21;
     int* ptr = &x;
     mystery(&ptr, y);
     cout << *ptr;
     return 0;
}
```

x
Address: 0x20

15

y, y2
Address: 0x40

4

ptr
Address: 0x80

0x40

What will be printed when this code executes?

```
void mystery(int** ptr2, int & y2) {
     y2 = 4;
     *ptr2 = &y2;
}
```

ptr2
Address: 0x120

0x80

```
int main() {
     int x = 15;
     int y = 21;
     int* ptr = &x;
     mystery(&ptr, y);
     cout << *ptr;
     return 0;
}
```

Output is:

4

x
Address: 0x20

15

y, y2
Address: 0x40

4

ptr
Address: 0x80

0x40

How can we modify this code?

```
void mystery(int** ptr2, int & y2) {
     y2 = 4;
     *ptr2 = &y2;
}


int main() {
     int x = 15;
     int y = 21;
     int* ptr = &x;
     mystery(&ptr, y);
     cout << *ptr;
     return 0;
}
```

How can we change the code so this line modifies ptr in main?

One solution: Pass in a pointer to the pointer!

Another solution: Reference to a pointer (int *& ptr2)