Pointers to Pointers

Pointers to pointers (1/4)

- Sometimes you want to have a procedure increment a variable?
- What gets printed?

```
void AddOne(int  x){
    x =  x + 1;
 }
int main() {
      int y = 5;
      AddOne( y);
      printf("y = %d\n", y);
}
```

Output:

y = 5

Pointers to pointers (2/4)

- Solved by passing in a pointer to our subroutine.
- Now what gets printed?

```
void AddOne(int *p){
    *p = *p + 1;
}
int main() {
      int y = 5;
      AddOne(&y);
      printf("y = %d\n", y);
}
```

Output:

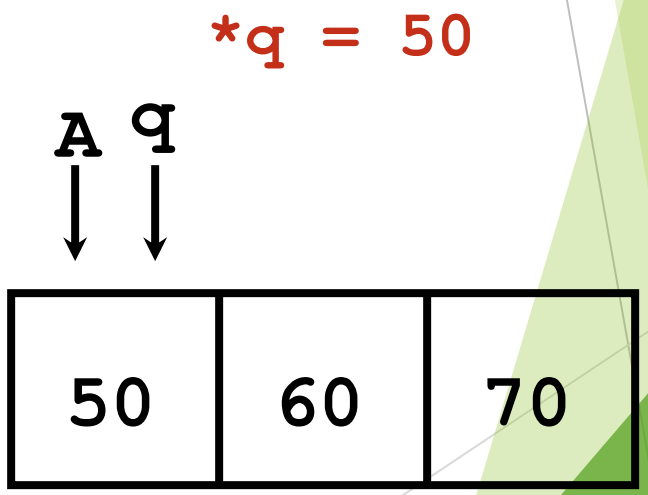
y = 6

Pointers to pointers (3/4)

- But what if what you want changed is a pointer?
- What gets printed?

```
void IncrementPtr(int  *p)
{    p =  p + 1;
}

int A[3] = {50, 60, 70};
int *q = A;
IncrementPtr( q);
printf("*q = %d\n", *q);
```



Pointers to pointers (4/4)

- Solution! Pass a pointer to a pointer, called a handle, declared as **h
- Now what gets printed?

```
void IncrementPtr(int **h) {
  *h = *h + 1;
 }

int A[3] = {50, 60, 70};
int *q = A;
IncrementPtr(&q);
printf("*q = %d\n", *q);
```

*q = 60