

Winter 2014 Midterm

Question 5

Legend



Enough space in memory to hold an integer value

Here, referred to with the variable name `x`

Here, also referred to with `*q` or by dereferencing `q` (what does `q` point at?). On the right side of `=`, `*q` will give the value stored in `x` (the integer); on the left side of `=`, `*q` will give the location of `x` (exactly how `x` is interpreted on the right or left side of `=`)

Enough space in memory to hold an address; also, the pointer is expected to point to an area in memory set aside to hold an integer value

Here, referred to with the variable name `q`

Here, the value stored in `q` is `&x` or the address of `x` (where in memory is `x` located?)

C program code (main)

```
int main() {  
  
    int x;  
    int y = 9;  
    int z;  
  
    int *q;  
    int *r;  
    int *s;  
  
    q = &x;  
    r = q;  
    *r = 3;  
    s = &y;  
    z = 5;  
  
    printf("xyz: %d, %d, %d\n", x, y, z);  
    printf("*: %d, %d, %d\n", *q, *r, *s);  
  
    unknown(&x, &y, &z);  
    unknown(q, r, s);  
  
    printf("xyz: %d, %d, %d\n", x, y, z);  
    printf("*: %d, %d, %d\n", *q, *r, *s);  
  
    unknown(&x, &y, &z);  
    unknown(q, r, s);  
  
    printf("xyz: %d, %d, %d\n", x, y, z);  
    printf("*: %d, %d, %d\n", *q, *r, *s);  
  
    return 0;  
}
```

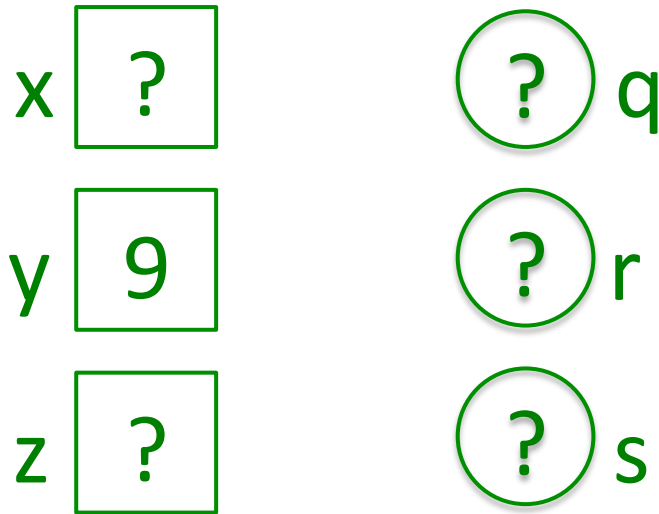
Local declarations in main

```
int main() {
```

```
    int x;  
    int y = 9;  
    int z;
```

```
    int *q;  
    int *r;  
    int *s;
```

```
    ...
```



Assignments in main

```
int main() {
```

```
    ...
```

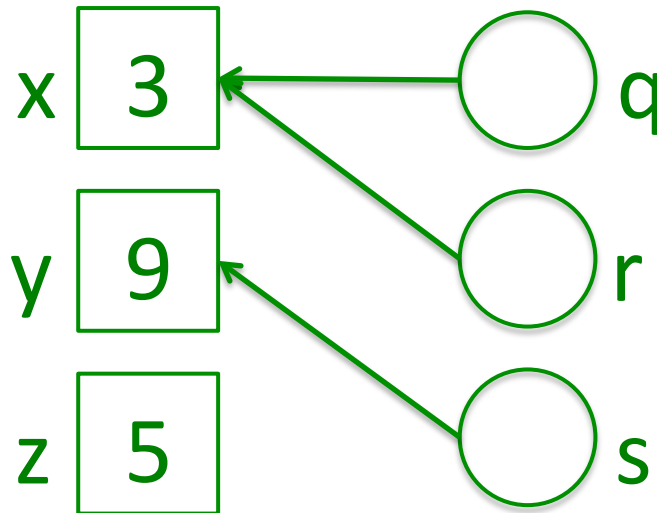
```
    q = &x;
```

```
    r = q;
```

```
    *r = 3;
```

```
    s = &y;
```

```
    z = 5;
```



First 2 printf's in main

```
int main() {
```

```
...
```

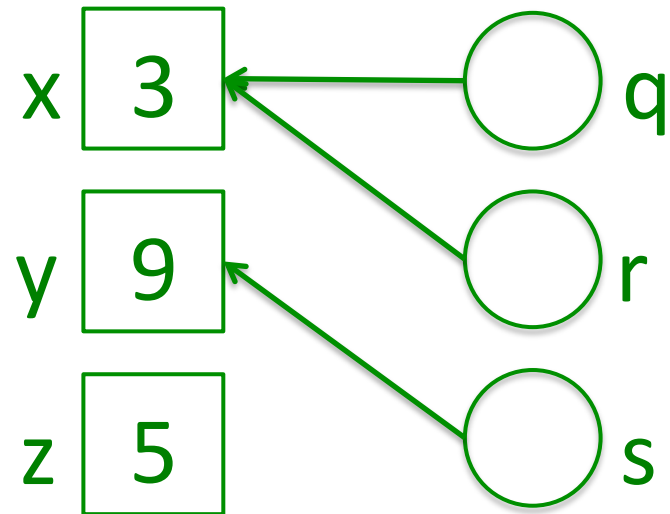
```
printf("xyz: %d, %d, %d\n", x, y, z);
```

```
printf("*: %d, %d, %d\n", *q, *r, *s);
```

```
...
```

XYZ: 3 9 5

*: 3 3 9



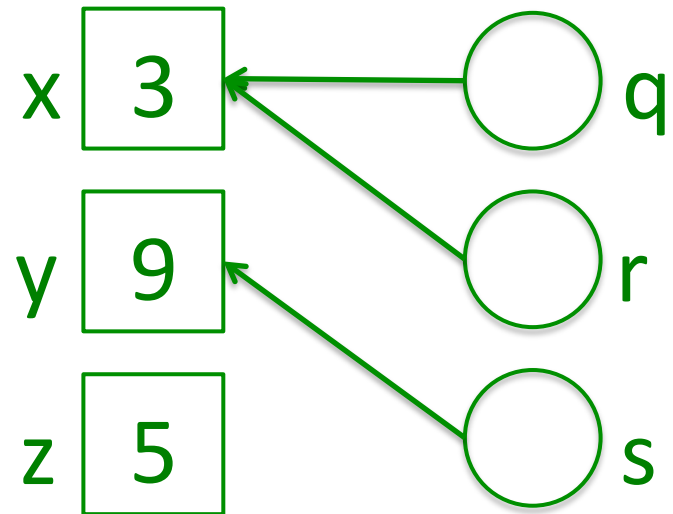
First call to unknown in main

```
int main() {
```

```
...
```

```
unknown(&x, &y, &z);
```

```
...
```



First call to unknown(&x, &y, &z)

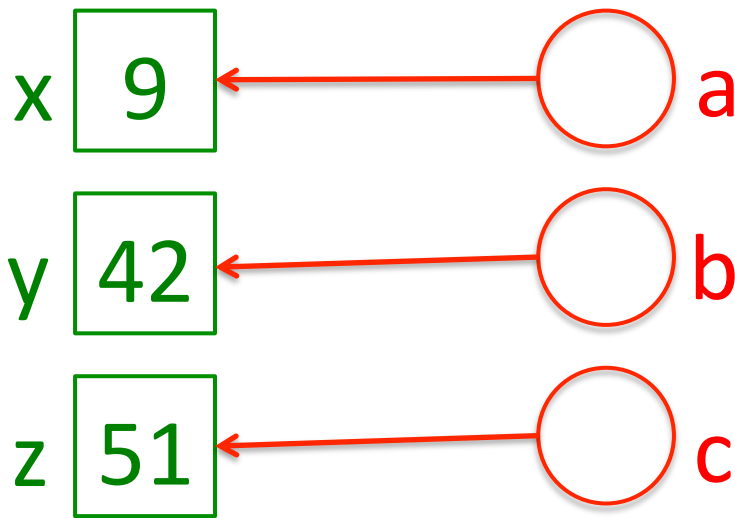
```
void unknown(int *a, int *b, int *c) {
```

```
    *a = *b;
```

```
    *b = 42;
```

```
    *c = *a + *b;
```

```
}
```



Second 2 printf's in main

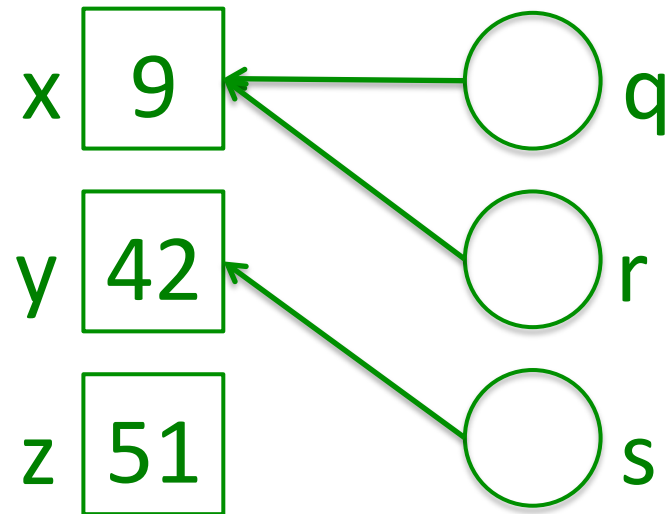
```
int main() {
```

```
...
```

```
printf("xyz: %d, %d, %d\n", x, y, z);
```

```
printf("*: %d, %d, %d\n", *q, *r, *s);
```

```
...
```



```
XYZ: 9 42 51
```

```
*: 9 9 42
```

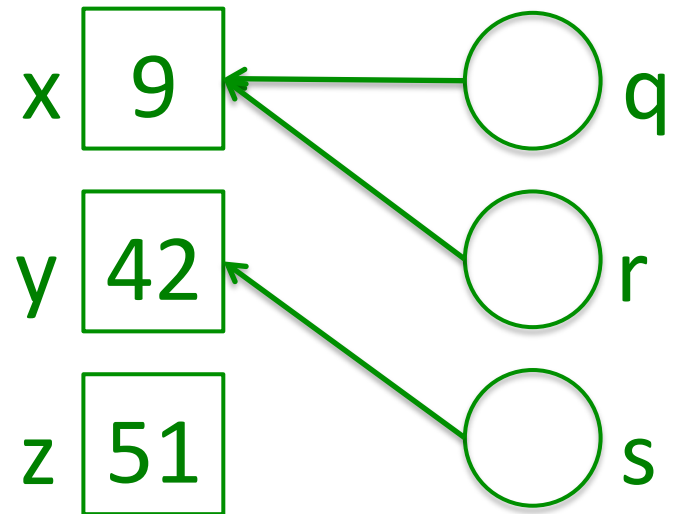
Second call to unknown in main

```
int main() {
```

```
...
```

```
unknown(q, r, s);
```

```
...
```



Second call to unknown(q, r, s)

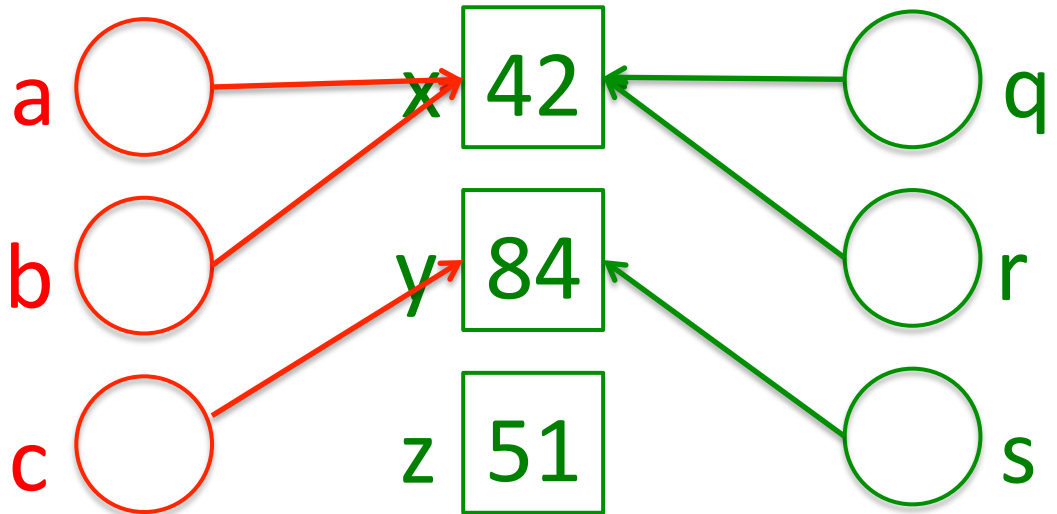
```
void unknown(int *a, int *b, int *c) {
```

```
    *a = *b;
```

```
    *b = 42;
```

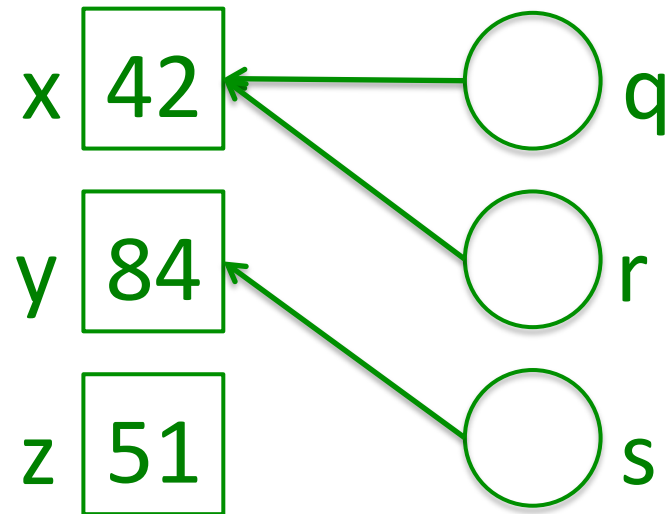
```
    *c = *a + *b;
```

```
}
```



Third 2 printf's in main

```
int main() {  
    ...  
    printf("xyz: %d, %d, %d\n", x, y, z);  
    printf("*: %d, %d, %d\n", *q, *r, *s);  
    return 0;  
}
```



XYZ: 42 84 51

*: 42 42 84