

EECS 280

Discussion 06: Feb 18, 2015

Agenda

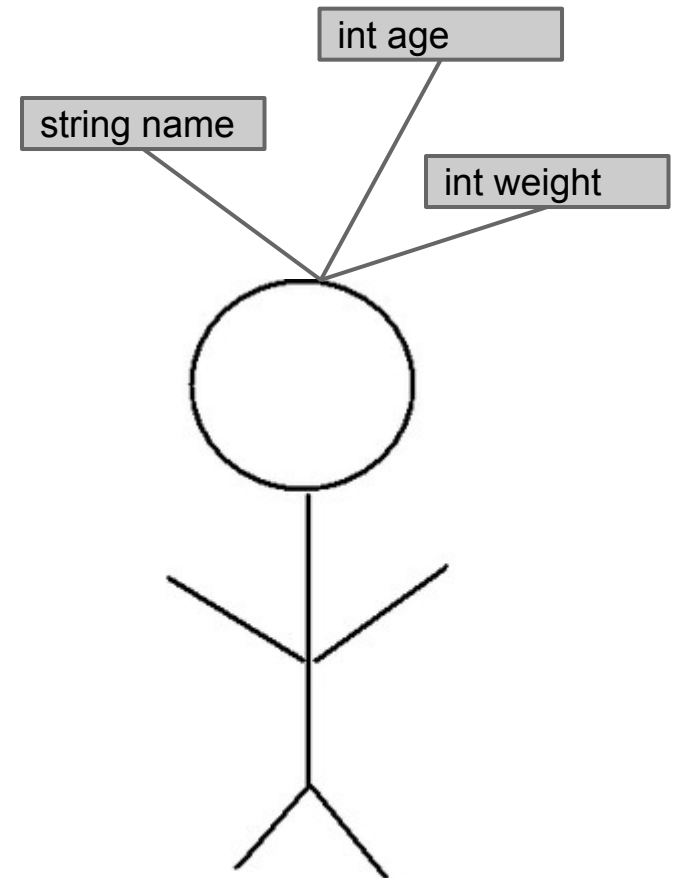
- Lab 06 due Friday 2/20
- Project 3 checkpoint due Thursday 2/19
- Midterm on Wednesday 2/25

- Brief review of lecture material
 - struct and class

- Work on Lab 06

struct and class

- Basic idea: encapsulate multiple pieces of data in one “collection”
- Analogy: real-world objects
- Each object is unique and has its own copy of the data



Accessing things within a struct

- We have a “plain old struct”?

```
void Point2D_print(Point2D p) {  
    int x = p.x;  
    int y = p.y;  
  
    cout << "(" << x << ", " << y << " )";  
}
```

- We have a pointer to a struct?

```
void Point2D_move(Point2D *p, double dx, double dy) {  
    p->x += dx;  
    p->y += dy;  
}
```

Accessing things within a struct

- We have an array of structs?

```
int Point2D_sum(Point2D points[], int num_points) {  
    int sum_x = 0;  
    int sum_y = 0;  
  
    for(int i = 0; i != num_points; i++) {  
        sum_x += points[i].x;  
        sum_y += points[i].y;  
    }  
  
    return sum_x*sum_x + sum_y+sum_y;  
}
```

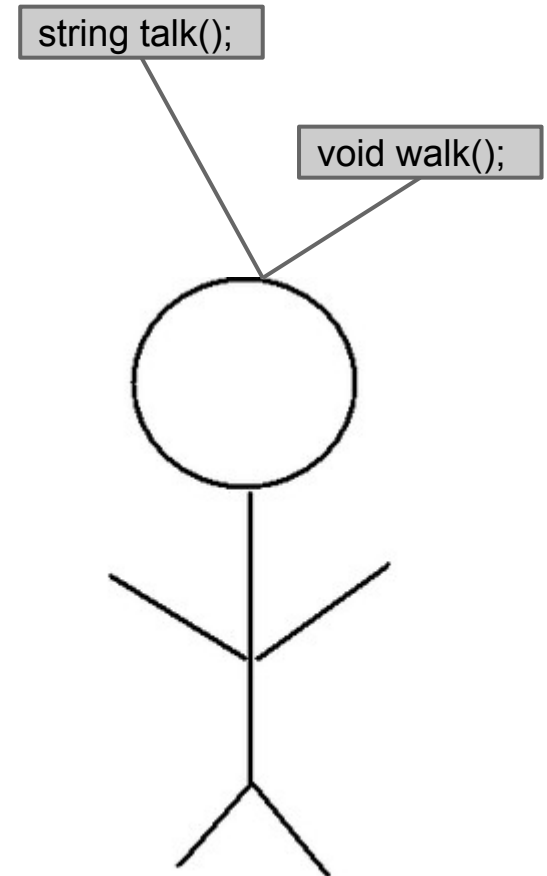
Accessing things within a struct

- We want to copy the value of a struct?

```
void Point2D_combine(Point2D *p1, const Point2D *p2) {  
    *p1 = *p2;  
}
```

struct and class

- Objects can also “do things”
- Analogy: performing an action
- Calling a function on one object doesn't affect any others



Lab 06

Goal

Practice using structs, classes, constructors, and member functions.

Tasks

1. Implement a function that modifies a struct
2. Add a constructor declaration and definition
3. Implement two class member functions