CS162: Introduction to 
Computer Science II

Structures

• Variables hold a single piece of
  “information”
  – Eg. an integer, a float, a character, a string
• Structures allow you to group logically
  connected variables into a single unit
Structures (Examples)

typedef struct Student {
    std::string name;
    int id;
    int num_courses_taken;
    std::string* courses;
};

typedef struct Customer {
    std::string name;
    int credit_card_number;
    int amount_spent;
};

Note: Struct names should start with a capital letter eg. Student

Structures

- Members of a struct are public by default
- Structs typically only contain member variables
- Structs rarely include member functions (but are allowed)
Initialization

There are two ways to initialize a struct:

1. Initialization list eg.
   Customer c = {"Bob", 9999, 100};

2. Using a constructor eg.
   Customer(std::string n, int cc, int amount) {
     name = n;
     cc = credit_card_number;
     amount_spent = amount;
   }

Accessing member variables

• Done with the dot operator eg.
   Customer c = {"Bob", 9999, 100};
   std::cout << c.name << std::endl;