

# Structures

CSE 130: Introduction to C Programming  
Spring 2005

1

## The Structure Type

- Makes it possible to aggregate components into a single, named variable
- Ex. a bank account contains an account #, a balance, an interest rate, etc.
- Structure components have individual names, and can be accessed individually
- A structure is a derived type

2

## Compare

```
int acctNumber;
```

```
char customerName[];
```

```
float balance;
```

```
int acctNumber;  
char customerName[];  
float balance;  
const float interestRate;
```

3

## Declaring a Structure

- Structure declarations begin with the keyword `struct`, followed by a tag name and a brace-enclosed list of components
- The tag name can be used to declare variables of the struct type
  - Variable type is “`struct <tag name>`”

4

## Structure Example

```
struct account ← tag name
```

```
{  
    long number;  
    float balance;  
    float interestRate;
```

```
};  
variable type
```

```
struct account myAcct;
```

5

## Structure Members

- Members of a structure can be accessed using the structure member (“.”) operator:

```
struct account a;  
a.balance = 1234.56;  
a.number = 8463745;
```

- Member names must be unique!
- Two different types of structure may have identical member names, though

6

## Structure Declarations

- We can combine a structure definition with variable declarations

```
struct card
{
    int value;
    char suit;
} c, deck[52];
```

7

## Structure Example 2a

```
struct fruit
{
    char name[15];
    int calories;
};

struct vegetable
{
    char name[15];
    int calories;
};
```

8

## Structure Example 2b

```
struct fruit a;
struct vegetable b;

a.calories = 35;
b.calories = 45;
```

9

## Another Example

```
struct student
{
    char *lastName;
    int studentID;
    char grade;
};
```

10

```
int fail(struct student class[])
{
    int i, count = 0;
    for (i = 0; i < CLASS_SIZE; i++)
        if (class[i].grade == 'F')
            count++;
    return count;
}
```

11

## Structure Initialization

- A structure variable can be followed by a list of constants contained within braces
  - remaining members are assigned 0
- Ex. struct card c = {12, 's'};
- Ex. struct fruit frt = {"plum", 150};
- We can also name members, as with arrays:  
struct card c = {.value = 5, .suit = 'd'};

12

## Structure Assignment

- If a and b are variables of the same structure type, we can write  
a = b;
- Each member of a is assigned the value of the corresponding value of b

13

## Passing Structures As Function Arguments

```
void assignValues(struct card c,  
                 int p, char s)  
{  
    c.value = p;  
    c.suit = s;  
}
```

14

## Passing Structures

```
void extractValues(struct card c,  
                 int p, char s)  
{  
    s = c.suit;  
    p = c.value;  
}
```

15

## Passing Structures

- When a structure is passed as an argument, it is copied (because of call-by-value)
- It is more efficient to pass the address of the structure instead

16

## Next Time

- Strings
- Pointers

17