Pointers and Structs

ITSC 2181: Introduction to Computer Systems
UNC Charlotte
College of Computing and Informatics

COLLEGE OF COMPUTING
AND INFORMATICS

structs Can Contain Pointers

```
struct person {
    char *name;
    …
} person1;

person1.name = “Donna”;
printf(“Name is %s\n”, person1.name);
char initial = *person1.name;
```

Are parentheses needed? No

Be careful when assigning string values from another function. Use the **strcpy()** or the **strncpy()** function.

Pointers **to** Structs

```
struct person {
    …
} person1, *p;

p = &person1;

(*p).name = “Donna”;
(*p).height = 65;
printf(“Name is %s\n”, (*p).name);
char initial = *(*p).name;
printf(“Height is %d\n”, (*p).height );
```

(see **`struct_pointer1.c`** in *Code samples and Demonstrations* in *Canvas*)

☠ ***common source of bugs*** ☠

failure to use parenthesis around `(*p).m`

Are parentheses needed?

Yes!

COLLEGE OF COMPUTING AND INFORMATICS

A New Operator: `->`

- Unfortunately, `*p.height != (*p).height`

  the value pointed to by the member `p.height`

  the height of the person pointed to by `p`

- A new operator (for convenience):
  `(*a).b` can be replaced by `a->b`

```
…
p = &person1;

p->name = “Donna”;
p->height = 65;
printf(“Name is %s\n”, p->name);
char initial = *p->name;
printf(“Height is %d\n”, p->height);
```

☠ ***common source of bugs*** ☠
failure to use parenthesis around `(*p).m`

(see `struct_pointer2.c` in *Code samples and Demonstrations* in *Canvas*)

What does `*` dereference?

COLLEGE OF COMPUTING AND INFORMATICS

A New Operator… (cont’d)

- How about **pointer** to a **`struct`** containing **pointer** to a **`struct`** containing…? No problem!

```
struct person {
     …
     struct person *father;
     struct person *mother;
} persons[100], *p;
p = &persons[1];
p->father = &persons[22];
p->mother = &persons[45];

if (p->father->age >= 65)
     …
printf(“Mother: %s\n”, p->mother->name);
```

Parentheses needed?

References

- S. J. Matthews, T. Newhall and K. C. Webb, *Dive into Systems*, Version 1.2. Free online textbook, available at: https://diveintosystems.org/book/
- K. N. King, *C Programming: A Modern Approach*, 2nd Edition. W. W. Norton & Company. 2008.
- D.S. Malik, *C++ Programming: From Problem Analysis to Program Design*, Seventh Edition. Cengage Learning. 2014.