# Programming with C I

**Fangtian Zhong**
**CSCI 112**

**Gianforte School of Computing**
**Norm Asbjornson College of Engineering**
**E-mail: fangtian.zhong@montana.edu**

2024.03.22

# Functions Whose Result Values are structured

- A function that computes a structured result can be modeled on a function computing a simple result.

- A local variable of the structure type can be allocated, filled with the desired data, and returned as the function result.

- The function does not return the address of the structure as it would with an array result.

- Rather, it returns the values of all components.

```
/*
 * Gets and returns a planet_t structure
 */
planet_t
get_planet (void)
{
        planet_t planet;

        scanf("%s%1f%d%1f%1f", planet.name,
                                &planet.diameter,
                                &planet.moons,
                                &planet.orbit_time,
                                &planet.rotation_time;
        return (planet);
}
```

# Problem Solving with Structure Types

**abstract data type (ADT**

- a data type combined with a set of basic operations

**Figure Data Type planet_t and Basic Operations**

| scan_planet | print_planet |
|---|---|
| planet_t | |
| | . . . |
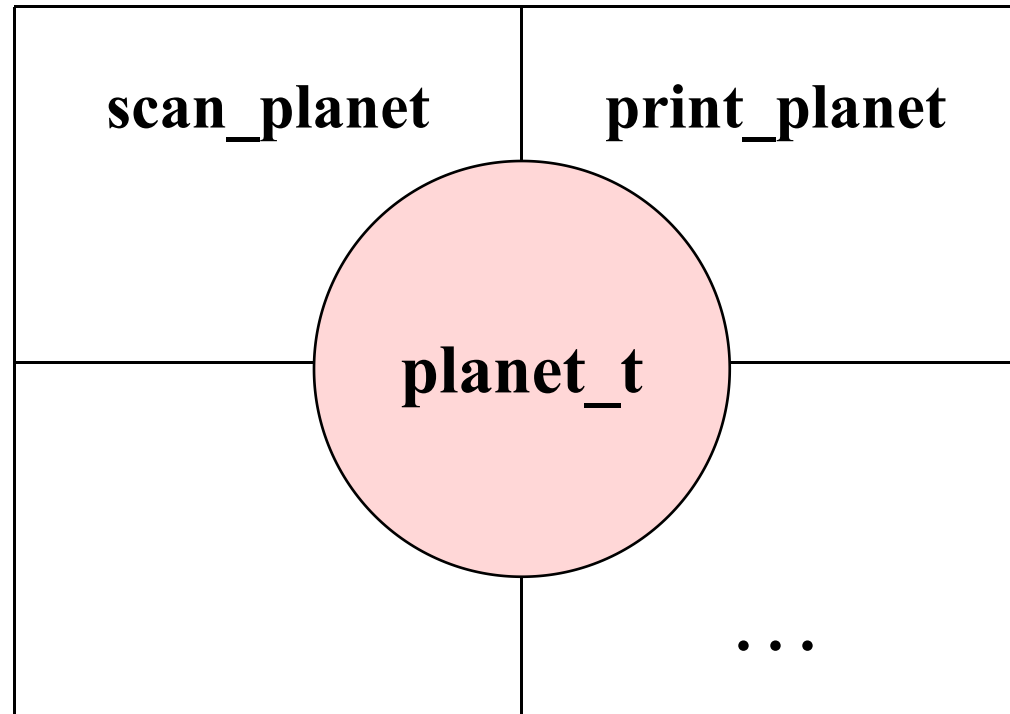
# Header files: defining the interface

#include<stdio.h>
versus
#include"class.h"

- ▶ **Angle brackets versus quotes tells compiler where to look for the file**

- ▶ **Gets copied in by preprocessor and then compiled in the .c file**

- ▶ **A .h file is never in the compile command**

**gcc –o exe –Wall program.c**

# Header guards

▸ **We don't want to include headers multiple times, but they may reference one another**

▸ **Solution: header guards**

```
#ifndef FILENAME_H

#define FILENAME_H

/* ... Declarations here ... */

#endif
```
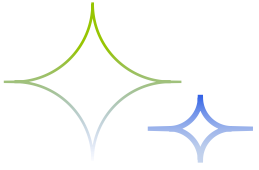
# THE END

**Fangtian Zhong**
**CSCI 112**

Gianforte School of Computing
Norm Asbjornson College of Engineering
E-mail: fangtian.zhong@montana.edu

2024.03.22