Introduction to Programming I

Session 1

MR. LESTER A. NACU

Department of Computer Science
College of Engineering
University of the Philippines, Diliman
SY 2011-2012 1S

Outline

- Basic C Program Structure
- Compiling and Running Program
- Variables and Data Types
- Standard I/O
- Operators and Expressions

Basic C Program Structure

```
#include <stdio.h>

int main()
{
    /*
       instructions
       goes here
    */
    return 0;
}
```

preprocessor directive

main function

the function body enclosed in “{“ and “}”

The Main Function

- First function the computer executes in the program
- More about functions soon!

Hello World!

```
#include <stdio.h>

int main()
{
    printf("Hello World");
    return 0;
}
```

from `printf()` to `0101100111`

- Open terminal and go to the directory (folder) where the source code is saved.
  - Use change directory **`cd`** command to navigate
  - Use list contents of directory **`ls`** command to print files in current directory
- Compile: **`gcc <filename>.c`**
- Run: **`./a.out`**

Comments

```
/*
   myfirstcprogram.c
   by Juan de la Cruz

   My first C program.
*/
#include <stdio.h>

int main ()
{
   printf("Hello world"); //prints hello world
   return 0;
}
```

Variable

- similar to containers
- reusable
- can hold a specific type of data
- has a definite size
  - may vary across different machines
- has a unique identifier that corresponds to a location in computer's memory
- has a value

Variable

Variable Declaration

SYNTAX:

<***data type***> <***identifier***>;

```
int    i_data;
float  f_data;
char   c_data1, c_data2;
```

Variable

Primitive Data Type

- `int` – integer (e.g. -45, 0, 2304)
- `float` – real number (e.g. -2123.4, -2.0, 0.0, 3.14)
- `char` – character (e.g. `'a'`,`'d'`,`'A'`,`'3'`,`'&'`,`' '`,`'\n'`)
- and many more!

```
int   i_data;
float f_data;
char  c_data1, c_data2;
```

Variable

Identifier

- any string of letters, digits, and underscore
- cannot start with a digit
- case-sensitive
- cannot use reserved words

```
int   i_data;
float f_data;
char  c_data1, c_data2;
```

Variable

Identifier

- any string of letters, digits, and underscore
- cannot start with a digit
- case-sensitive
- cannot use reserved words

INVALID

```
5star   return  first-name  Data.1
```

CASE-SENSITIVE

```
firstname   firstName   FIRSTNAME
```

Variable

Assigning Value

- use assignment operator =

SYNTAX:

```
<identifier> = <value>;
```

```
int   i_data; // declare variable
i_data = 5; // assign value
float f_data = 3.14; // declare and assign
```

Variables in Action

```
#include <stdio.h>

int main()
{
    int my_int = 4, your_int, another_int = 2;
    float realn = 2.0;
    char letter = 'c';

    your_int = 9231;
    realn = 94.1232;

    return 0;
}
```

Standard I/O

- from `stdio.h`
- using `printf()` for formatted output
- using `scanf()` for input

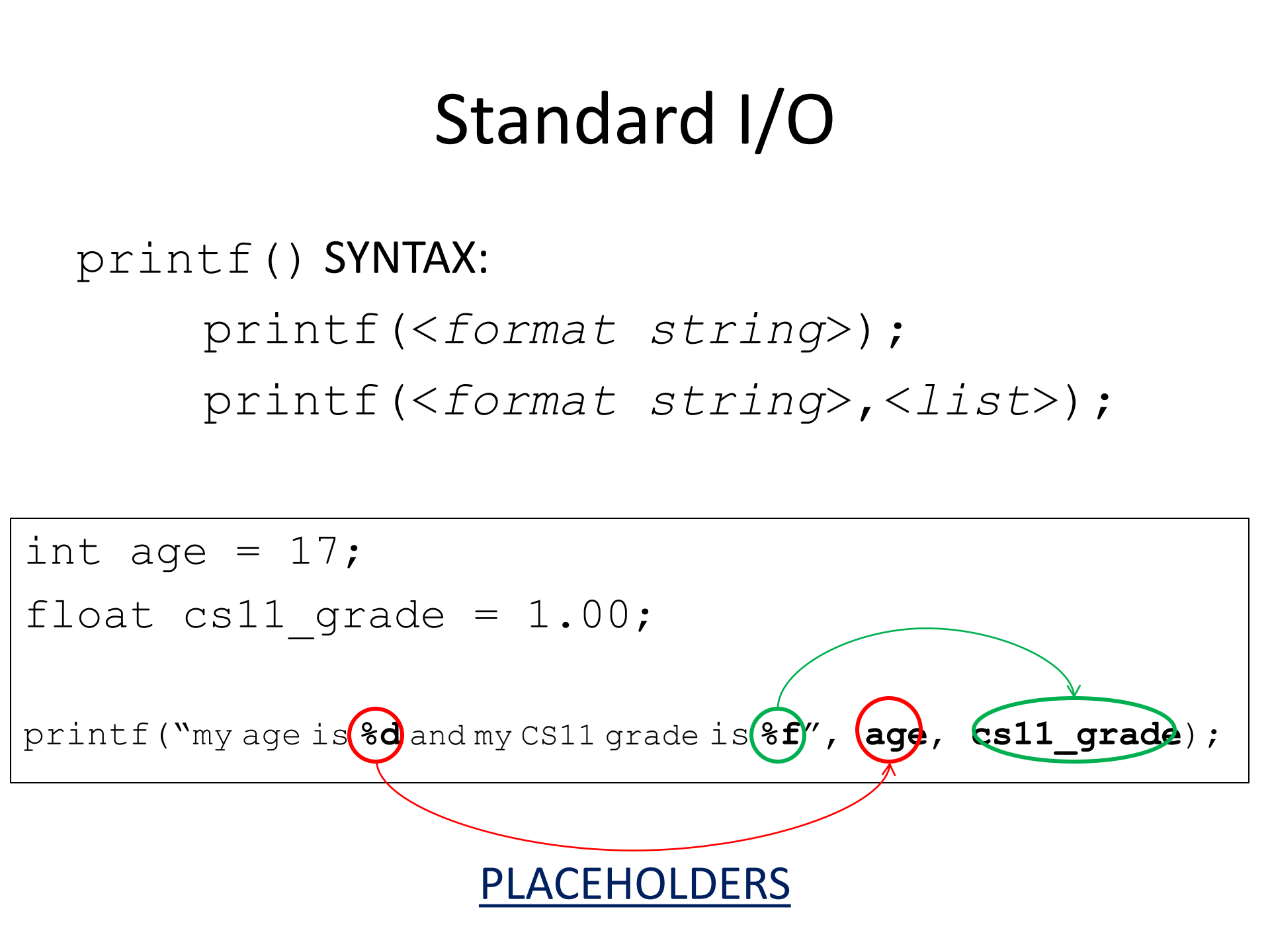
Standard I/O

printf() SYNTAX:

```
printf(<format string>);
printf(<format string>,<list>);
```

```
int age = 17;
float cs11_grade = 1.00;

printf("my age is %d and my CS11 grade is %f", age, cs11_grade);
```



Standard I/O

```
int age = 17;
float cs11_grade = 1.00;

printf("my age is %d and my CS11 grade is %f", age, cs11_grade);
```

output

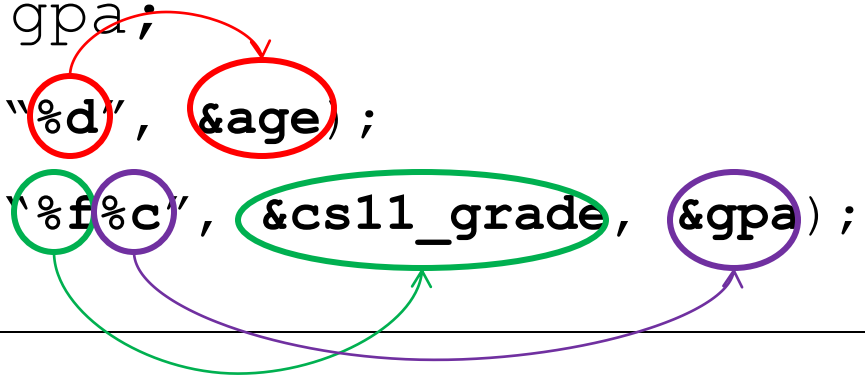
```
my age is 17 and my CS11 grade is 1.00
```

Standard I/O

`scanf()` SYNTAX:

`scanf(<format string>,<list>);`

```
int age;
float cs11_grade;
char gpa;
scanf("%d", &age);
scanf("%f%c", &cs11_grade, &gpa);
```



PLACEHOLDERS

Standard I/O

```
int age;
float cs11_grade;
char gpa;
scanf("%d", &age);
scanf("%f%c", &cs11_grade, &gpa);
```

Let’s talk!

```
#include <stdio.h>

int main()
{
    int age;
    printf(“How old are you? ”);
    scanf(“%d”, &age);
    printf(“\nYou are only %d?!”, age);
    return 0;
}
```

Operators

- Arithmetic
  - i.e. `*`, `/` , `%`, `+`, `-`
  - `/` and `%` are undefined when divisor is `0`
  - `3/2` evaluates to `1` (why?)
- Logic
  - i.e. `!`, `&&`, `||`
- Comparison
  - i.e. `>`, `<`, `==`, `>=`, `<=`, `!=`

Operators

- What's `4+3*2`?
- expression is first evaluated before the result is assigned to a variable
  - e.g. `int x = 345.9/829;`

  (what is the value of x? why?)
  - e.g. `float x = 4.5 + 1/2 * 8.6;`

  (what is the value of x? why?)

Don’t Forget!

- For any questions, concerns or appointment for consultation email me at mr.nacu@gmail.com or visit me at room 316.

Thank you