16.216: ECE Application Programming
Summer 2014

Lecture 2: Key Questions
May 22, 2014

1. What are the basic binary arithmetic operators supported by C?

2. Explain the modulus operator (%).

3. What determines the type of a binary operation's result?

4. What is the difference between division of integers and floating-point types?

5. Explain the operation of the unary negation operator (e.g., `-x`).

6. **Example:** Evaluate each of the following expressions, including the type (`int` or `double`) in your answer.

a. `19/3`

b. `3/19`

c. `19%3`

d. `3%19`

e. `5 + 7/2`

f. `5.0 + 7/2`

g. `5 + 7.0/2`

h. `5 * 3 % 3 / 6 + 14 + 10 / 2`

i. `5 * (3 % 3) / 6 + 14.0 + 10/3`

7. Describe the use of `printf()` to print numeric values and characters.

8. **Example:** Show the output of each of the following short programs:

a.
```c
#include <stdio.h>
void main()
{
    int i=2, j=3, k, m;
    k = j * i;
    m = i + j;
    printf("%d %d %d %d\n", i, j, k, m);
}
```

b.
```c
#include <stdio.h>
void main() {
    double f, g;
    f = 1.0 / 4.0;
    g = f * 20;
    printf("f = %lf,\ng = %lf\n", f, g);
}
```

c.
```c
#include <stdio.h>
void main() {
    int a = 5, b = 2;
    printf("Output%doesn't%dmake%dsense", a, b, a + b);
}
```

9. Describe the use of `scanf()` for reading input values into variables.

10. How does `scanf()` handle whitespace and other characters in format string?

11. **Example:** Assume you have the following variables: `int i; double d; char c;`
    If your program contained each of the following calls to scanf(), what values would be read
    into the appropriate variables, given user input?

a. Input: `34 5.7`
    `scanf("%d%lf", &i, &d)`

b. Input: `34 5.7`
    `scanf("%d        %lf", &i, &d)`

c. Input: `34 5.7`
    `scanf("%lf%d", &d, &i)`

d. Input: `34 5.7`
    `scanf("%d%c", &i, &c)`

e. Input: `34 5.7`
    `scanf("%d %c", &i, &c)`

12. Explain how to set the field width, alignment, and padding characters for values printed using `printf()`.

13. Explain how to set the precision of a value printed using `printf()`, and what the precision means for the different data types.

14. **Example:** Assume `int x = 123; float y = 4.56; double z = 7.89991;`
    What does each of the following lines print?

a. `printf("%4d %5f %6lf\n", x, y, z);`

b. `printf("%.4d %.4f %.4lf\n", x, y, z);`

c. `printf("%08d %-7.1f %+-4.1lf !\n", x, y, z);`

15. **Example:** Write a short code sequence to do each of the following:
a.   Print three integers—x, y, and z
   * Use field widths of 10, 20, and 30, respectively
   * Put an extra space between each field
   * Show the signs of all values and left justify them

b.   Print four doubles—d1, d2, d3, d4
   * Use field widths of 7 for all values
   * Put an extra space between each field
   * Show 1, 2, 3, and 4 places after the decimal point, respectively

c.   Given three variables—int w, p; double var;
   * Read values for w and p from the input
   * Print var using field width w and precision p

16. Describe the basic elements of a flowchart.

17. Design a flowchart to solve the following:
   - Prompt a user to enter four numbers on a single line, which represent the contents of a 2x2 array
   - After reading the values, your program should print the matrix represented by these values
     - For example, if the user enters "1 2 3 4", print:
       ```
       1   2
       3   4
       ```
     - Assume all values have the same number of digits
   - Also, calculate the matrix discriminant and print it on a separate line
     - In the example above, discriminant = (1x4) - (2x3) = 4-6 = -2

18. Convert the flowchart you wrote into a C program.

19. Explain the useful features of a debugger.

Note: At this point, we'll run through the use of the Visual Studio debugger; feel free to use this space to take notes on that demonstration.