EECE.2160: ECE Application Programming

Summer 2016

Lecture 3: Key Questions
May 19, 2016

1. Explain the basic form of an `if` statement.

2. Describe how the expression in `if (<expression>)` is evaluated and show how conditions are evaluated, including multiple conditions in the same expression.

3. Describe how the statement—the actual code to be executed if the condition is true—is written for an `if` statement.

4. Show how multiple if statements can be nested together (`if/else if/else`).

5. **Example:** What does the following code print?

```
int main() {
    int x = 3;
    int y = 7;

    if (x > 2)
        x = x - 2;
    else
        x = x + 2;

    if ((y % 2) == 1)
    {
        y = -x;
        if ((x != 0) && (y != -1))
           y = 0;
    }
    printf("x = %d, y = %d\n", x, y);
    return 0;
}
```

6. Discuss how to use `if` statements to check that a value falls within a desired range.

7. **Example:** Write a short code sequence that does each of the following:

a. Given `int x`, check its value. If `x` is more than 5 and less than or equal to 10, print `x`

b. Prompt for and read temperature as input (type `double`). If temp is 90 or higher, print "It's too hot!" If temp is 32 or lower, print "It's freezing!" In all other cases, print "It's okay"

c. Read 3 `int` values and print error if input problem
   - If fewer than 3 values read, print error message with number of values
   - Example: `Error: only 2 inputs read correctly`

8. Describe the basic format of a `switch` statement, including its general usage, the use of `case` and `default`, and the use of the `break` statement.

9. Describe a situation in which you might not want to use a `break` statement at the end of a given `case`.

10. **Example:** Given the code below:

```
int main() {
      char grd;

      printf("Enter Letter Grade: ");
      scanf("%c",&grd);
      printf("You are ");

      switch (grd)      {
      case 'A' :
            printf("excellent\n");
            break;
      case 'B' :
            printf("good\n");
            break;
      case 'C' :
            printf("average\n");
            break;
      case 'D' :
            printf("poor\n");
            break;
      case 'F' :
            printf("failing\n");
            break;
      default :
            printf("incapable of reading directions\n");
            break;
      }
      return 0;
}
```

What does the program print if the user inputs:

a. A

b. B+

c. c

d. X

11. How could we easily change each case to recognize both upper and lowercase inputs?

12. Describe the basic elements of a flowchart.

13. 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

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

15. 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.