1. When you write an `if`-block for idiotproofing, does the `exit` statement belong **before** the `if`-block, **inside** the `if`-block or **after** the `if`-block? Therefore, should the `exit` statement be indented **less than** the `if` statement, **the same as** the `if` statement, or **more than** the `if` statement?

2. **DEBUG** the following program. If you aren’t confident of your answer, type in, compile and run the C program to test it.

```c
#include <stdio.h>
int main ()
{
    int input_value1, input_value2;
    int aggregate = 0;

    printf("Input two integers:\n");
    scanf("%d %d", &input_value1, &input_value2);
    printf("Before\n");
    if (input_value1 < input_value2) ||
        (input_value1 > input_value2) {
        printf("First\n");
        aggregate = input_value1 + input_value2;
    } /* if (input_value1 < input_value2) || ... */
    printf("After\n");
    printf("aggregate = %d\n", aggregate);
} /* main */
```
3. The Kelvin temperature scale is very similar to the Celsius temperature scale, except that zero degrees Kelvin is absolute zero, the lowest physically conceivable temperature. Zero degrees Kelvin is -273.16 degrees Celsius.\footnote{http://www.usatoday.com/weather/wtempcf.htm}

Write a program that prompts for and inputs a temperature in degrees Kelvin, then idiot-proofs, then calculates the associated temperature in degrees Celsius, then outputs the temperature in degrees Celsius.

You \textbf{DON'T} have to use comments. Otherwise, all rules for Programming Projects (through PP#4) apply.
4. **DESCRIBE THE CONDITION** of a *while* loop. (‘The condition is a ...’)

5. Are the properties of the condition of a *while* loop the same as, or different from, the properties of the condition of an *if* block?

6. **WHAT ARE THE STEPS** that describe the execution of a *while* loop?

   (a)

   (b)

   (c)

7. **HOW** does a *while* loop **DIFFER** from an *if* block?
8. For each of these kinds of statements, mark **CAN** if it can appear in the body of a *while* loop, and mark **CANNOT** if it cannot appear in the body of a *while* loop. **EXPLAIN.**

(a) A named constant declaration

(b) A variable declaration

(c) A `printf` statement

(d) A `scanf` statement

(e) An assignment statement

(f) A `exit` statement

(g) An `if` block

(h) A `while` loop
9. **TRACE** the example program on slides 23 - 25 of the lecture packet titled “while Loop Lesson,” using the input values shown on slides 26 - 27. Your trace should show the following variables: `users_number`, `users_distance`, `users_last_distance` and `correct_number_hasnt Been_input`, but in the trace you can abbreviate their names as `un`, `ud`, `uld` and `cnhbi`, respectively.
10. **DRAW A FLOWCHART** for the Infinite Loop program on slide 15 of the lecture slide packet titled “**while Loop Lesson**.”

If you use **ANY** resources other than Dr. Neeman, the TAs (Gaur, Gheibi, Reynolds, Sadri), the course textbook or the materials posted on the course webpage, you **MUST** reference them on the quiz. **THIS INCLUDES CLASSMATES, FRIENDS, PROFESSORS, ONLINE RESOURCES, ETC.**