1. **5.8** (p. 94) The following `if` statement is unnecessarily complicated. Simplify it as much as possible. (*Hint:* The entire statement can be replaced by a single assignment.)

   ```
   if (age >=13)
      if (age <=19)
         teenage = true
      else
         teenage = false
   else if (age < 13)
      teenage = false
   ```

2. **5.11** (p. 97) Write a program that asks the user for a two-digit number, and then prints the English word for the number:

   ```
   Enter a two-digit number: 45
   you entered the number forty-five.
   ```

   Hint: Break the number into two digits. Use one `switch` statement to print the word for the first digit ("twenty", "thirty", and so forth). Use a second `switch` statement to print the word for the second digit. Don't forget that the numbers between 11 and 19 require special treatment.

3. **6.1-3** (p. 121) What output does each of the following program fragment produce?

   (a)
   ```
   i = 1;
   while (i <= 128) {
     printf("%d ", i);
     i *= 2;
   }
   ```
   (b)
   ```
   i = 9384;
   do {
     printf("%d ", i);
     i /= 10;
   } while (i > 0);
   ```
   (c)
   ```
   i = 5;
   j = i - 1;
   for (; i > 0, j > 0; --i, j = i-1)
     printf("%d ", i);
   ```

4. What output does each of the following flowcharts produce? Write a program fragment in C for each of the flowcharts, and submit the codes online.
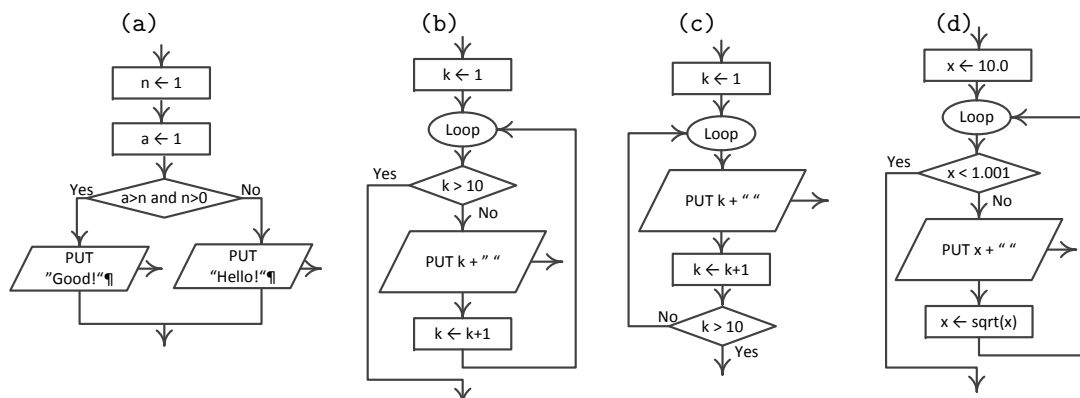
5. **6.11** (p. 124) The value of the mathematical constant ϵ can be expressed as an infinite series:

   ϵ = 1 + 1/1! + 2/2! + 3/3! + ...

   Write a program that approximate ϵ by computing the value of

   ϵ = 1 + 1/1! + 2/2! + 3/3! + ... + n/n!

   where n is an integer entered by the user. Save and submit the program as `a3_epsilon0.c`.

6. **6.12**[*] (p. 124) The attached flowchart `a3_epsilon.rap` provides a modified solution to Q3.5. It allows for continuous addition of terms until the current term becomes less than a small (floating-point) number ε entered by the user. Write an equivalent program in C, and save and submit the program as `a3_epsilon1.c`.