

16.216: ECE Application Programming

Fall 2014

Lecture 9: Key Questions September 22, 2014

1. Explain the usage and basic structure of a `while` loop.

2. **Example:** What does each of the following short programs print?

a. `x = 7;`
 `while (x < 10)`
 `{`
 `printf("%d ",x);`
 `x = x + 1;`
 `}`

b. `x = 7;`
 `while (x < 3)`
 `{`
 `printf("%d ",x);`
 `x = x + 1;`
 `}`

```
int main() {
    int i;                // Number to square
    int iSquared;          // Square of the number
    printf(" i           i^2\n"); // Column headings

    // Compute and display the squares of numbers 0 to 10
    // Use a field width of 2 to print i and 10 to print i^2
    //   with no extra space between the fields

    return 0;
}
```

- 4

5. What is the difference between a `while` loop and a `do-while` loop?

6. Show the difference between the outputs of the loops below

```
x = 7;
do {
    printf("%d",x);
    x = x + 1;
} while ( x < 3 );
```

```
x = 7;
while ( x < 3 )
{
    printf("%d",x);
    x = x + 1;
}
```

7. Recall the example for using a while loop with a sentinel value in the grade average program and show that loop written as a do-while loop.

```
/*
 * while2.c
 * Adapted from earlier solution by Prof. George Cheney
 * 16.216: ECE Application Programming
 * ECE Dept., UMass Lowell
 *
 * PURPOSE: Read list of grades from keyboard and compute average
 *
 * DEMONSTRATES: A counting loop with a flexible limit
 */

#include <stdio.h>

int main()
{
    int numGrades;      // Requested # of grades
    int gradeCount;     // Counts # of grades processed so far

    double grade;       // An individual grade to be processed
    double gradeSum;    // Running total
    double avgGrade;    // Average grade

    // Prompt for and read # of grades
    printf("How many grades? ");
    scanf("%d", &numGrades);

    // Prompt user to enter grades
    printf("Enter %d grades:\n", numGrades);

    // Initialize loop
    gradeSum = 0;
    gradeCount = 0;

    // Repeatedly read grades until limit is reached
    while (gradeCount < numGrades) {
        scanf("%lf", &grade);      // Read grade and accumulate it
        gradeSum = gradeSum + grade;

        gradeCount = gradeCount + 1; // Increment grade count
    }

    // Compute and display the average
    avgGrade = gradeSum / numGrades;
    printf("Average grade = %1.11f\n", avgGrade);
    return 0;
}
```

```
/*
 * while3.c
 * Adapted from earlier solution by Prof. George Cheney
 * 16.216: ECE Application Programming
 * ECE Dept., UMass Lowell
 *
 * PURPOSE: Read list of grades from keyboard and compute average.
 * Program will run until user enters invalid grade (something
 * outside the range  $0 \leq \text{grade} \leq 100$ ), which is taken as
 * signal that
 *
 * DEMONSTRATES: Loop terminated on sentinel value
 */

#include <stdio.h>

int main()
{
    int gradeCount;    // Counts # of grades processed so far

    double grade;      // An individual grade to be processed
    double gradeSum;    // Running total
    double avgGrade;    // Average grade

    char enterGrade;

    // Initialize loop
    gradeSum = 0;
    gradeCount = 0;

    // Prompt for and read first grade
    /*printf("Enter grade: ");
    scanf("%lf", &grade);*/

    // Continue reading/accumulating grades until invalid value entered
    do {
        printf("Enter grade: ");    // Prompt for and read next grade
        scanf("%lf", &grade);

        // If invalid grade entered, leave loop
        if ((grade > 100) || (grade < 0))
            break;

        gradeSum = gradeSum + grade;    // Accumulate grade
        gradeCount = gradeCount + 1;    // Increment grade count
    } while ((grade >= 0.0) && (grade <= 100.0));

    if (gradeCount == 0)    // No grades entered
        printf("No valid grades entered\n");

    // Compute and display the average
    else {
        avgGrade = gradeSum / gradeCount;
        printf("Average grade = %1.1lf\n", avgGrade);
    }

    return 0;
}
```