

Chapter 4 Written Homework

R4.1 Provide trace tables for these loops.

- a. `int i = 0; int j = 10; int n = 0;`
`while (i < j) { i++; j--; n++; }`

0	10	0
1	9	1
2	8	2
3	7	3
4	6	4
5	5	5
- b. `int i = 0; int j = 0; int n = 0;`
`while (i < 10) { i++; n = n + i + j; j++; }`

0	0	0
1	1	1
2	2	4
3	3	9
4	4	16
5	5	25
6	6	36
7	7	49
8	8	64
9	9	81
- c. `int i = 10; int j = 0; int n = 0;`
`while (i > 0) { i--; j++; n = n + i - j; }`

10	0	0
9	1	8
8	2	14
7	3	18
6	4	20
5	5	20
4	6	18
3	7	14
2	8	8
1	9	0
- d. `int i = 0; int j = 10; int n = 0;`
`while (i != j) { i = i + 2; j = j - 2; n++; }`
i is never equal to j.

R4.2 What do these loops print?

- for (int i = 1; i < 10; i++) { cout << i << " "; }
1 2 3 4 5 6 7 8 9
- for (int i = 1; i < 10; i += 2) { cout << i << " "; }
1 3 5 7 9
- for (int i = 10; i > 1; i--) { cout << i << " "; }
10 9 8 7 6 5 4 3 2
- for (int i = 0; i < 10; i++) { cout << i << " "; }
0 1 2 3 4 5 6 7 8 9
- for (int i = 1; i < 10; i = i * 2) { cout << i << " "; }
1 2 4 8
- for (int i = 1; i < 10; i++) { if (i % 2 == 0) { cout << i << " "; } }
2 4 6 8

R4.3 What is an infinite loop? On your computer, how can you terminate a program that executes an infinite loop?

An infinite loop occurs when the conditions are coded in a way where the program will never exit the loop, meaning that it runs on forever. You can terminate a program with an infinite loop by either coding it correctly the first time or just shutting down your terminal.

R4.5 Write a program trace for the pseudocode in Exercise P4.9, assuming the input values are 47 -2 -5 0.

First = true

Minimum = 47 first = false

2 less than minimum, minimum = 2

50 greater than minimum

2 printed as minimum

R4.7 How often do the following loops execute? Assume that i is not changed in the loop body.

- for (int i = 1; i <= 10; i++) ...
10
- for (int i = 0; i < 10; i++) ...
10
- for (int i = 10; i > 0; i--) ...
10
- for (int i = -10; i <= 10; i++) ...
21
- for (int i = 10; i >= 0; i++) ...
11
- for (int i = -10; i <= 10; i = i + 2) ...
11
- for (int i = -10; i <= 10; i = i + 3) ...
7

R4.8 Write pseudocode for a program that prints a calendar such as the following:

```
Su M T W Th F Sa
    1  2  3  4
 5  6  7  8  9 10 11
12 13 14 15 16 17 18
19 20 21 22 23 24 25
26 27 28 29 30 31
```

Cout << first line of days

Cout << second line

Counter to check if until 7, Boolean for newline

For loop that increments by 1 from 5 to 31

Cout inside forloop to print out dates and adequate spacing

R4.10 Write pseudocode for a program that reads a sequence of student records and prints the total score for each student. Each record has the student's first and last name, followed by a sequence of test scores and a sentinel of -1. The sequence is terminated by the word END. Here is a sample sequence:

```
Harry Morgan 94 71 86 95 -1
Sally Lin 99 98 100 95 90 -1
END
```

Provide a trace table for this sample input.

Read in the first name and last name, judging by where the spaces break the string

Boolean/if statement to check for -1

Declare variable for sum of scores, variable for input

Have a cin, and have that cin register to the input variable

While there is no -1, add input to the sum of scores

R4.12 Rewrite the following do/while loop into a while loop.

```
int n;
cin >> n;
double x = 0;
double s;
do
{
    s = 1.0 / (1 + n * n);
    n++;
    x = x + s;
}
while (s > 0.01);
```

```
int n;
cin >> n;
double x = 0;
double s;
while ( s > 0.01)
{
    s = 1.0 / (1 + n * n);
```

```

    n++;
    x = x + s;
}

```

R4.13 Provide trace tables of the following loops.

- a. `int s = 1; int n = 1; while (s < 10) { s = s + n; } n++;`

1	1
2	1
3	1
4	1
5	1
6	1
7	1
8	1
9	1
9	2
- b. `int s = 1; for (int n = 1; n < 5; n++) { s = s + n; }`

1	1
2	2
3	4
4	7
- c. `int s = 1; int n = 1; do { s = s + n; n++; } while (s < 10 * n);`

1	1
2	2
3	4
4	7
5	11

R4.17 Add a storyboard panel for the conversion program in Section 4.6 on page 154 that shows a scenario where a user enters incompatible units.

Noticeable groans from the user and tester. “Did you try cin.fail() yet?”

Please enter amount rounded to nearest int:

Seventy-Five Dollars and 20 cents

Wrong Type. Please Try Again.

R4.18 In Section 4.6, we decided to show users a list of all valid units in the prompt. If the program supports many more units, this approach is unworkable. Give a story board panel that

illustrates an alternate approach: If the user enters an unknown unit, a list of all known units is shown.

Please enter amount rounded to nearest int:

Seventy - Five Dollars and 20 cents

Wrong Type. Please Try Again.

Acceptable inputs:

- 17
- 20
- 150

R4.19 Change the storyboards in Section 4.6 to support a menu that asks users whether they want to convert units, see program help, or quit the program. The menu should be displayed at the beginning of the program, when a sequence of values has been converted, and when an error is displayed.

Convert units

help

quit program

Please enter amount rounded to nearest int:

Seventy - Five Dollars and 20 cents

Wrong Type. Please Try Again.

R4.23 The nested loops `for (int i = 1; i <= height; i++) { for (int j = 1; j <= width; j++) { cout << "*" ; } cout << endl; }` display a rectangle of a given width and height, such as `**** *
**** *
**** *`. Write a single for loop that displays the same rectangle.

```
for(int i = 0; i < 3; i++)  
{  
    cout << "****\n";  
}
```

R4.24 Suppose you design an educational game to teach children how to read a clock. How do you generate random values for the hours and minutes?

I would make sure that the hours would never exceed 12, and that minutes would never exceed 60.