**KIT506 Programming Practical 3: Repeat actions for a number of steps**

[ [↶ Close all sections](https://mylo.utas.edu.au/content/enforced/214790-AW_EAI_17S2_15468_0_0_0_1_1/kit506_pracs/KIT506_Prac_03.html?d2lSessionVal=dx0bynATNq1YR9mA2AHARswOX&ou=214790&d2l_body_type=3) ] Only visible sections will be included when printing this document.

**Aims:**

* to understand multiway branching with **switch**; and
* to use **for** loops to repeat an action a given number of times.

**∨ 1 Reading Exercises**

**1.1 Nested ifs**

Trace the following code to determine the final value of result:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14 | int x = 10;  int y = 5;  int result = 0;  **if** (x % y == 0) {  result = 1;  **if** (x < y) {  result = 2;  } **else** {  result = 3;  }  } **else** {  result = 4;  } |

Enable Solutions

If we were to trace the value of all variables in this program we would produce the following tracing table:

| **Line** | **x** | **y** | **result** |
| --- | --- | --- | --- |
| 1 | [**Solution**](javascript:void(0);)  10 |  |  |
| 2 | [**Solution**](javascript:void(0);)  10 | [**Solution**](javascript:void(0);)  5 |  |
| 3 | [**Solution**](javascript:void(0);)  10 | [**Solution**](javascript:void(0);)  5 | [**Solution**](javascript:void(0);)  0 |
| 6 | [**Solution**](javascript:void(0);)  10 | [**Solution**](javascript:void(0);)  5 | [**Solution**](javascript:void(0);)  1 |
| 10 | [**Solution**](javascript:void(0);)  10 | [**Solution**](javascript:void(0);)  5 | [**Solution**](javascript:void(0);)  3 |

[table contents](https://mylo.utas.edu.au/content/enforced/214790-AW_EAI_17S2_15468_0_0_0_1_1/kit506_pracs/KIT506_Prac_03.html?d2lSessionVal=dx0bynATNq1YR9mA2AHARswOX&ou=214790&d2l_body_type=3)

Which tells us the the final value of result is 3.

To work this out we can trace the program to the first decision point at line 5 (x % y == 0, which asks ‘is x evenly divisible by y?’) to determine that 10 *is* evenly divisible by 5 and so the if branch will be executed. This updates the value of resultto 1, but the program keeps going…

[**Solution**](javascript:void(0);)

**1.2 Switch**

Without creating a tracing table (you won’t need it), determine the output of this code if the user enters the text *monday*when prompted:

string day;

string message;

Console.Write("What day is it today? ");

day = Console.ReadLine();

**switch** (day.ToLower())

{

**case** "sunday": message = "Time for the weekly shopping"; **break**;

**case** "monday": message = "I hate Mondays - Garfield"; **break**;

**case** "tuesday": message = "Only four more days to go"; **break**;

**case** "wednesday": message = "Middle of the week!"; **break**;

**case** "thursday": message = "Will there be thunder later?"; **break**;

**case** "friday": message = "TGIF!"; **break**;

**case** "saturday": message = "It's the weekend!"; **break**;

**default**: message = "I don't understand what day of the week '" + day + "' is."; **break**;

}

Console.WriteLine(message);

What if the user were to enter *TueSDaY*?

Finally, what if the user entered *onsdag*, which is the name of Wednesday in Swedish, Norwegian and Danish?

**Tip:** You can control a **switch** statement with integers, characters, strings (as above) and enumerated types.

[**Solution**](javascript:void(0);)

If the user enters *Monday* then the program will output

I hate Mondays - Garfield

What if the user were to enter *TueSDaY*?

[**Solution**](javascript:void(0);)

Only four more days to go

By calling ToLower() on the string the user has entered we guarantee that we can match English day names to the lower case values in the switch’s cases.

Finally, what if the user entered *onsdag*, which is the name of Wednesday in Swedish, Norwegian and Danish?

[**Solution**](javascript:void(0);)

I don't understand what day of the week 'onsdag' is.

**Tip:** You can control a **switch** statement with integers, characters, strings (as above) and enumerated types.

**∨ 2 Repeating actions for a given number of times**

**2.1 Repeat *n* times**

Create a new program and in the Main method write a **for**-loop that prints 20 asterisks (\*). Before you write it answer the following, keeping in mind the structure of the loop:

**for** (*initialisation*; *boolean expression to test*; *update*)  
{  
    *body*  
}

* What is the *initialisation*?
* What is the *test*?
* What is the *update*?
* What is the loop *body*?
* What is the *initialisation*?

[**Solution**](javascript:void(0);)

Array indices start at zero, so we should initialise the loop variable (call it n) to that: int n = 0.

* What is the *test*?

[**Solution**](javascript:void(0);)

We want to stop when n is beyond the end of the array, which we can read using counts.Length, so i < counts.Length

* What is the *update*?

[**Solution**](javascript:void(0);)

Since we’re visiting each position in the array, we should step by 1: n++

* What is the loop *body*?

[**Solution**](javascript:void(0);)

We want to print the current *position*, followed by |, followed by the value at that position in the array, which we can access via counts[n], so Console.WriteLine(n + " | " + counts[n]) is suitable.

[**Solution**](javascript:void(0);)

**2.2 For each *index* in an array or List**

One of the common uses for a **for**-loop is for iterating of the indices (positions) in structures such as Lists and arrays. *Temporarily* comment out the loop code you wrote before (wrap it in an opening /\* and closing \*/) as you’ll use it again next, and write some new code before it.

Add the following at the start of the Main method. It declares and initialises an array of integer values:

int[] counts = {1, 0, 5, 7, 4, 2, 1};

Diagrammatically, counts looks like:

| **index** | **0** | **1** | **2** | **3** | **4** | **5** | **6** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| value | 1 | 0 | 5 | 7 | 4 | 2 | 1 |

**Task:** Now write a **for**-loop that will display each value in the array counts, one per line, prefixed by its position and a pipe symbol |. In other words, we want the output to look like:

0 | 1

1 | 0

2 | 5

3 | 7

4 | 4

5 | 2

6 | 1

* What is the *initialisation*? Int I = 0
* What is the *test*? I<=6
* What is the *update*? I++
* What is the loop *body*? {…}

int[] counts={1,0,5,7,4,2,1};

for (int i = 0; i <= 6; i++)

{

Console.WriteLine("{0} | {1}", i, counts[i]);

}

**2.3 (Optional challenge) Horizontal histogram**

What if we want to display a text-based histogram (bar chart) instead of just the values from the array? Can you think of how you might combine the [first loop](https://mylo.utas.edu.au/content/enforced/214790-AW_EAI_17S2_15468_0_0_0_1_1/kit506_pracs/KIT506_Prac_03.html?d2lSessionVal=dx0bynATNq1YR9mA2AHARswOX&ou=214790&d2l_body_type=3#stars) with the [second one](https://mylo.utas.edu.au/content/enforced/214790-AW_EAI_17S2_15468_0_0_0_1_1/kit506_pracs/KIT506_Prac_03.html?d2lSessionVal=dx0bynATNq1YR9mA2AHARswOX&ou=214790&d2l_body_type=3#arrays) to produce output like this?

0 | \*

1 |

2 | \*\*\*\*\*

3 | \*\*\*\*\*\*\*

4 | \*\*\*\*

5 | \*\*

6 | \*

int[] counts={1,0,5,7,4,2,1};

for (int i = 0; i <= 6; i++)

{

Console.Write("{0} | ",i);

for (int j = 1; j <= counts[i]; j++)

{

Console.Write('\*');

}

Console.WriteLine();

}

**∨ 3 Next week: test + while loops**

At the start of next week’s practical class you’ll do a 20-minute online quiz on programming skills up to this point, which will contribute to your final grade. After that you’ll do some practical exercises involving another form of loop, the **while** loop.