

cc-practice-do

Table of Contents

- [1. README](#)
- [2. TODO Identify yourself](#)
- [3. TODO Counting down](#)
- [4. TODO Limits](#)
- [5. TODO Summing numbers](#)

1. README

- Practice workbook for `do...while` loops in C.

2. TODO Identify yourself

Replace the placeholder `[yourName]` in the header of this file by your name and save the file (`C-x C-s`).

3. TODO Counting down

1. Rewrite the counting program with a `do` instead of a `while` loop. The output should look like this:

```
T minus 10 and counting
T minus 9 and counting
T minus 8 and counting
T minus 7 and counting
T minus 6 and counting
T minus 5 and counting
T minus 4 and counting
T minus 3 and counting
T minus 2 and counting
T minus 1 and counting
```

2. Set the counter initially to 10, and use the *postfix compound decrement* operator `i--`.

3. Begin by writing the pseudocode for the `do...while` loop.

— PSEUDOCODE —

— PUT THE CODE HERE —

— Solution —

- Pseudocode:

```
do {
  print index
```

```
    decrement index by 1
} while index is greater than 0
```

- Code: this is the concise version with the decrement operator inside the function call.

```
int i = 3;

do {
    printf("T minus %d and counting\n", i--);
} while (i > 0);
```

```
T minus 3 and counting
T minus 2 and counting
T minus 1 and counting
```

— END SOLUTION —

4. **TODO** Limits

1. What if `i >= 2147483647`?
2. Run the loop for `i = 2147483647`, for `i = 2147483648` and for `i = 2147483649`.
3. Instead of printing each number, define a counter `int count=0` and count up while `i` counts down, then print count **after** the loop to see how many times the loop was run.
4. For fun, while running it, open a terminal and type the command `top`. Now you will see the CPU at work while the loop executes.

— SOLUTION —

```
int i = 2147483648;
int count = 0;

do {
    i--;
    count++;
} while (i > 0);
printf("%d\n", count);
```

```
-2147483648
```

— END SOLUTION —

5. **TODO** Summing numbers

1. Rewrite the summing numbers program with a `do` instead of a `while` loop. Try to get your output to look exactly like this:

```
: Enter integers (0 to terminate): 18 23 71 5 1 0
: The sum is 118
```

2. How does the pseudocode look like?

— PSEUDOCODE —

— PUT THE CODE HERE —

— SOLUTION —

Pseudo code:

```
do {  
  add input integer to sum  
  scan next input integer  
} while input integer is not 0
```

Input file:

```
echo "-18 23 71 -5 1 0" > ./src/sum_input_solution  
cat ./src/sum_input_solution
```

Code:

```
int n=0, sum = 0;  
  
printf("Enter integers (0 to terminate): ");  
  
do {  
  sum += n;  
  scanf("%d", &n); printf("%d ", n);  
} while ( n != 0 );  
  
printf("\nThe sum is %d\n", sum);
```

```
Enter integers (0 to terminate): -18 23 71 -5 1 0  
The sum is 72
```

Created: 2022-06-16 Thu 11:40