

# Tutorial Sheet 3

## Exercise 1

Re-write the program in Listing 6.1 to include just a single call to `scanf()`.

### Listing 6.1:

```
#include <stdio.h>

int main(void)
{
    long num;
    long sum = 0L; /* initialize sum to zero */
    int status;

    printf("Please enter an integer to be summed (q to quit): ");
    status = scanf("%ld", &num);
    while (status == 1)
    {
        sum = sum + num;
        printf("Please enter next integer (q to quit): ");
        status = scanf("%ld", &num);
    }
    printf("Those integers sum to %ld.\n", sum);

    return 0;
}
```

## Exercise 2

Write a program to compute `n!` using a `for` loop.

---

## Exercise 3

Write a program that performs a PIN code check, giving end-users a maximum of 5 attempts before locking them out.

---

## Exercise 4

Write a program that tabulates the multiplication tables for numbers 1 through 10 over 10 rows, i.e.:

```
1  2  3  4  5  6  7  8  9  10
2  4  6  8 10 12 14 16 18 20
...

```

---

## Exercise 5

Write a program that accepts as input a sequence of up to 20 non-unique numbers (between 1 and 100, in ascending order) and then computes the mean, mode, and median.

---

## Exercise 6

Write a program that calculates your taxes based on the following rates and incentives:

- The first €10,000 are taxed at **18%**.
  - The next €8,000 are taxed at **20%**.
  - Any remaining income is taxed at **25%**.
  - If you're working in the **ICT industry**, a rebate of **5%** on the total tax bill applies.
  - If you collect old electronic equipment for green disposal, the first tax bracket is raised to **€15,000**.
- 

## Exercise 7

**Write a program that processes an input text stream and takes the following actions:**

- Warns about words that are longer than 10 characters and do not include a hyphen.
- Warns about words containing uppercase characters beyond the first one.
- Warns about words starting with a non-alphabetic character.
- Displays the total number of errors in the form of:
  - “1 error found”
  - “n errors found”
  - “No errors found”

**Test your program on specifically crafted test input files as well as random text files found on your machine.**

---

## Exercise 8

**Modify the program from Exercise 7 so that it also:**

- Warns about punctuation preceded by a space.
  - Warns about repeated spaces.
- 

## Exercise 9

**Write a program that processes an input stream of real numbers delimited by whitespace, and:**

- Copies them to an output file as long as the numbers fall within a preset range.
- Takes care of validating all inputs while attempting to deal gracefully with non-compliant inputs.

*Note: You may use functions such as `fopen()`, `fscanf()`, `fprintf()`, and `fclose()` for file operations. Ensure that you handle file opening errors and input validation appropriately.*

---

## Exercise 10

Write a program that repeatedly displays the following menu options:

1. Add items to shopping cart
2. Show current total
3. Check out
4. Cancel session
- q. Quit

- **Option 1:** Asks the end-user to choose from a list of preset shop items (displayed as code - name - price) and quantity, keeping a running total for the current shopping session.
- **Option 2:** Displays the current total.
- **Option 3:** Displays the itemized bill of purchased products & total, then starts a new session.
- **Option 4:** Cancels the current session and starts afresh.
- **Option q:** Terminates the program immediately.

**Note:** Option selections for **3**, **4**, and **q** should be confirmed by the user.