

# POPRO – lab 4

## Implementacja prostych zadań algorytmicznych.

Autor: Daniel Paczesny, Łukasz Dąbała, Sebastian Kozłowski, Krystian Ignasiak

### 1 Tematyka

Laboratorium 4 z przedmiotu POPRO dotyczy implementacji w języku C prostych zadań algorytmicznych.

### 2 Przygotowanie do zajęć laboratoryjnych

W ramach przygotowania do laboratorium warto zapoznać się z materiałami z zakresu podstawowych typów danych w języku C, funkcji do pobierania danych ze standardowego wejścia i funkcji wypisujących dane na standardowe wyjście oraz instrukcji sterujących.

1. <https://www.studytonight.com/c/c-input-output-function.php>
2. <http://www.cs.cornell.edu/courses/cs2022/2011sp/lectures/lect09.pdf>

### 3 Sprawozdanie

Do zadań 1 – 3 w repozytorium powinien znaleźć się plik z rozszerzeniem `.c`, w którym znajduje się rozwiązanie danego problemu. Dodatkowo, należy stworzyć plik sprawozdania, w którym znajdują się odpowiedzi na zadane pytania.

### 4 Przebieg laboratorium

#### 4.1 Zadanie 1 – pobieranie danych z linii komend i wypisywanie na standardowe wyjście (1 pkt.)

Napisz program *bow\_tie.c*, który pobiera z linii polecenia 2 argumenty  $N$  oraz  $M$ , które są odpowiednio liczbą kolumn i wierszy. Program powinien drukować na standardowym wyjściu obraz podobny do muchy – elementu garderoby męskiej – podobnej do poniższego obrazka.

```

* . . . . *
** . . **
*** . ***
*****
*** . ***
** . . **
* . . . . *

```

Zastosuj dwie pętle *for* oraz jedną konstrukcję *if-else*. Czy dla każdej liczby  $N$  oraz  $M$  taki obrazek będzie wyglądał poprawnie?

## 4.2 Zadanie 2 – Instrukcje sterujące i przetwarzanie ciągu znaków (1 pkt.)

Napisz program *shuffle.c*, który pobiera z linii polecenia kolejne litery (małe lub wielkie). Liczba liter do pobrania jest określana jako stała w programie. Zadaniem programu jest *pseudolosowe* pomieszenie odczytanych liter. Wykryj, czy otrzymany rezultat mieszania liter nie spowodował przypadkowego ich posortowania w dowolnym porządku (rosnąco lub malejąco).

Na standardowym wyjściu program powinien wypisać: jaka jest maksymalna liczba liter jaką program może przetworzyć, jaki został odczytany ciąg liter wprowadzony przez użytkownika, pomieszany ciąg znaków oraz informację, czy wynik mieszania nie spowodował posortowania ciągu liter.

Zabezpiecz program przed brakiem lub błędnymi danymi wejściowymi.

## 4.3 Zadanie 3 – Tablice (2 pkt.)

Treść zadania zostanie podana przez prowadzącego na początku zajęć.

Napisz program, który jako parametr wywołania przyjmuje ciąg znaków dowolnej długości, a następnie zlicza wystąpienia cyfr.

Po zliczeniu wystąpień, program powinien wypisać pary *cyfra: liczba\_wystąpień*, o ile liczba wystąpień jest większa od 0.

Przykład:

**Input:** ab4b45.Hk

**Output:**

4: 2

5: 1