

Контролно 1
по Увод в алгоритмите и структури от данни за 10 клас

Време за работа: 80 минути

При грешки при компилация за задачата ще получите 0 точки!

```
$ gcc -Wall --pedantic --std=c11 filename.c
```

Задача 1

Имплементирайте следната функция:

```
unsigned int count_10(short data);
```

която като аргумент получава двубайтово число, а като резултат връща броя срещания на последователността **10** в подаденото число.

Демонстрирайте нейната работа в main.

Примери:

$11_{10} (00000000\ 00001011_2) \Rightarrow 1$

$3_{10} (00000000\ 00000011_2) \Rightarrow 0$

$21601_{10} (01010100\ 01100001_2) \Rightarrow 4$

Задача 2

Имплементирайте следната функция:

```
void sort_most_common(int *array, unsigned int size);
```

която като аргументи получава масив от числа и броя елементи в него. Функцията да намери най-често срещаната стойност от масива и да премести всички нейни повторения в **края**. Ако няколко различни стойности имат равенство за най-често срещаната то алгоритъмът да избира първата срещната такава.

Демонстрирайте нейната работа в main.

Обяснете в коментар сложността на алгоритъма ви.

Примери:

$[1, 3, 5, 3, 7] \Rightarrow [1, 5, 7, 3, 3]$

$[3, 4, 2, 4, 3] \Rightarrow [4, 2, 4, 3, 3]$

Задача 3

Дефинирайте двусвързан списък, в който чрез указатели се пазят едновременно реда на добавяне и сортирания ред на елементите.

- нови елементи се добавят в началото на списъка, съответно head винаги сочи последно добавения елемент
- всеки елемент има указатели за предишен и следващ, които се насочват според реда им на добавяне
- освен това всеки елементи има указатели за предишен и следващ във възходящ ред
- при добавяне на нов елемент всички тези указатели се насочват към правилните елементи

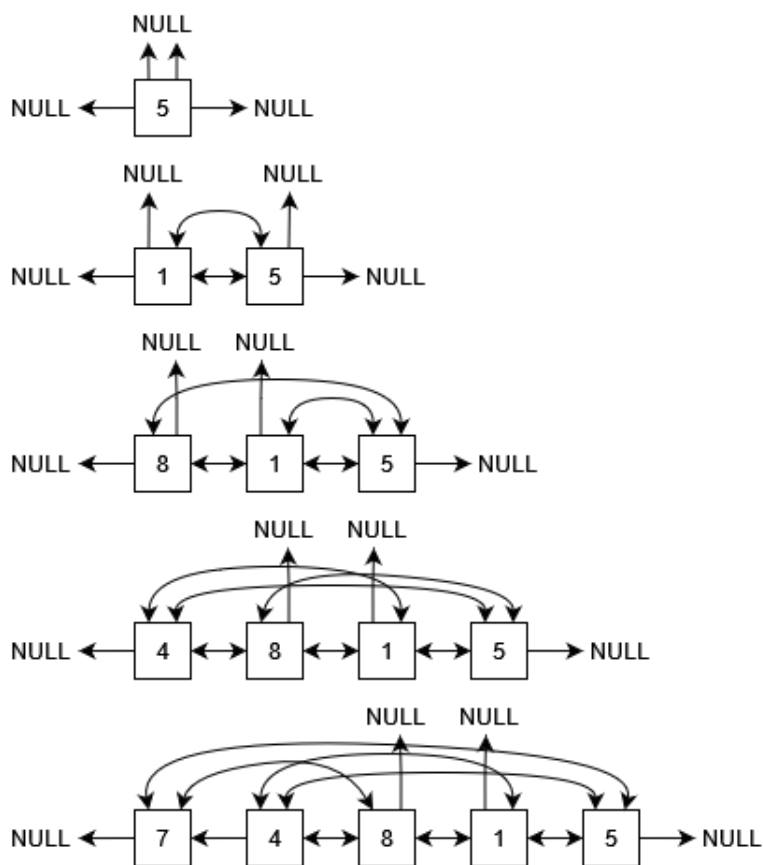
```
struct node_t* init_node(int value);
struct node_t* add_node(struct node_t *list, int value);
void print_as_added(struct node_t *list);
void print_as_sorted(struct node_t *list);
```

Да се демонстрира работата на функциите в main.
Обяснете в коментар сложността на функцията add_node.

Примери:

При добавяне на 5, 1, 8, 4, 7

Списъкът трябва да има такъв вид:



Където:

- хоризонталните стрелки показват указателите за реда на добавяне
- вертикалните стрелки показват указателите за реда на сортиране
- указателят за начало на списъка(head) винаги сочи най-левия елемент