Домашнее задание №1

## Общие замечания.

* При решении задач нужно использовать только те возможности языка Go, которые были рассмотрены на тренинге на момент выдачи домашнего задания (некоторые исключения могут оговариваться особо в условии конкретных задач).
* Так как создание пользовательских функций на момент выдачи **этого** задания ещё не изучалось, весь основной код размещайте в функции main.
* Для ввода и вывода данных используйте функции из пакета fmt – Print, Println, Scanln, Printf, Scanf (последние две – по желанию).
* Считайте, что пользователь всегда вводит данные в корректном формате. Ошибки ввода не обрабатывайте (пример – ситуацию, когда в задаче нужно ввести число, а пользователь ввёл произвольную строку, мы не рассматриваем).
* Решение каждой задачи помещайте в отдельный каталог вида Task\_NN, например – Task\_01 (см. примеры кода к лекциям).
* Все решения запакуйте в один zip-архив и отправьте по почте тренеру.

## Задача 1.

Программа запрашивает у пользователя натуральное число, меньшее 1\_000\_000\_000. Если пользователь ввёл число не из указанного диапазона, программа выводит строку «Input error» и останавливается. Иначе выводится самая большая цифра введённого числа.

## Задача 2.

Проверить, является ли введённое положительное целое число [палиндромом](https://ru.wikipedia.org/wiki/%D0%9F%D0%B0%D0%BB%D0%B8%D0%BD%D0%B4%D1%80%D0%BE%D0%BC), если записать его в троичной системе счисления.

## Задача 3.

Система оценок в США (в зависимости от процента от максимального количества баллов): **A** – 90-100%, **B** – 80-89%, **C** – 70-79%, **D** – 65-69%, **F** – 64% и ниже. Программа запрашивает максимальное количество баллов за одно задание, количество заданий, а затем – баллы за каждое задание. (Имеется в виду, что некий студент выполнил ряд заданий, и мы хотим оценить успеваемость этого студента). Программа выводит оценку в виде буквы. \*) Считаем, что ввод пользователя всегда корректен и его можно не проверять.

## Задача 4.

Программа запрашивает у пользователя два целых числа **a** и **b**. Затем программа выводит все положительные целые числа в диапазоне от **a** (включительно) до **b** (включительно), которые в своём двоичном представлении имеют ровно 4 единицы. Учтите, что пользователь может ввести число **a**, которое больше числа **b**, – в этом случае числа надо поменять местами.