

1. Написать функцию, принимающую отсортированный динамический массив целых чисел, его длину и значение искомого элемента и возвращающую индекс последнего элемента в массиве, имеющего искомое значение, методом, аналогичным двоичному поиску (просто найти двоичным поиском какой-то элемент, равный искомому, а затем циклом бежать от него до первого, имеющего то же значение, не годится, потому что таких элементов может быть слишком много). Если в массиве такого элемента нет, нужно вернуть «-1».
2. Ввести с консоли несколько строк. Количество строк заранее неизвестно (указатель на массив строк, поставить ограничение длины массива), признак конца ввода – EOF (end of file – строка “end” – можно определить директивой define). Распечатать введенные строки, сконкатенированные по 2 – первая с последней, вторая – с предпоследней и т.д. Использовать динамическое выделение памяти, можно использовать функции `strlen`, `strcpy`, `strcat`.

(`int strlen(char *s)` – возвращает длину строки без учета завершающего 0-символа.

`char *strcpy(char *s1, char *s2)` – копирует `s2` в `s1`, возвращает `s1`

`char *strcat(char *s1, char *s2)` – добавляет `s2` к `s1` в конец., последним символом добавляет `\0`, возвращает `s1`.)

3. Написать функцию, принимающую вещественное число x и два указателя f и g на функции типа `double(double)`, и возвращающую результат применения f к сумме и результата применения g к квадрату x