Задачи.

**1. Распечатать произвольное число в двоичной системе исчисления.**

*Описание.*

Написать функцию, принимающую на вход произвольное целочисленное значение и возвращающую строку, содержащую его представление в двоичной системе исчисления.

*Декларация.*

void NumberAsBinary(

char \* \_result, // строка с результатом

unsigned int \_number // произвольное число

)

*Ожидаемый результат.*

Например, для числа 13 данная функция должна вернуть строку "1101".

*Проверка работы.*

Решение должно демонстрировать работу функции на примере 10 случайных чисел в диапазоне 0 до 512.

**2. Сгенерировать неупорядоченный массив целых чисел от 1 до N.**

*Описание.*

Написать функцию, принимающую на вход произвольное целочисленное значение и возвращающую массив, содержащий последовательность чисел от 1 до этого значения, в котором каждое число последовательности находится в заведомо случайной позиции.

*Декларация.*

void GenerateRandomArray(

int \* \_result, // возвращаемый массив

unsigned int \_maxValue // максимальное число в последовательности

)

*Ожидаемый результат.*

Например, 2 последовательных запуска данной функции с параметром 7 могут сгенерировать последовательности вида

{2, 5, 7, 3, 4, 1, 6}

{5, 1, 3, 4, 6, 2, 7}

*Проверка работы.*

Решение должно демонстрировать работу функции на примере 4 случайных чисел в диапазоне от 5 до 15.

**3. Сортировка массива.**

*Описание.*

Написать функцию, принимающую в качестве параметра массив неупорядоченных целых чисел (сгенерированный с помощью функции из задачи 2) и выполняющую его сортировку.

*Декларация.*

void Sort(

int \* \_values, // сортируемый массив

unsigned int \_count // количество элементов в массиве

)

*Ожидаемый результат.*

Например, массив вида {2, 5, 7, 3, 4, 1, 6} будет преобразован данной функцией в {1, 2, 3, 4, 5, 6, 7}.

*Проверка работы.*

Решение должно демонстрировать работу функции на примере результатов, полученных при решении задачи 2 (последовательностей, сгенерированных для 4 случайных чисел в диапазоне от 5 до 15).

**4. Определить недостающее число в неупорядоченной последовательности целых чисел.**

*Описание.*

Написать функцию, принимающую в качестве параметра массив неупорядоченных целых чисел (сгенерированный с помощью функции из задачи 2) и определяющую значение последнего элемента в массиве без доступа к нему.

*Декларация.*

int GetMissingValue(

const int \* \_values, // неупорядоченный массив целых чисел от 1 до \_count

unsigned int \_count // количество элементов в массиве

)

*Ожидаемый результат.*

Например, для последовательности {5, 7, 8, 3, 1, 9, 2, 6, 4} данная функция должна вернуть 4.

*Проверка работы.*

Решение должно демонстрировать работу функций на примере результатов, полученных при решении задачи 2 (последовательностей, сгенерированных для 4 случайных чисел в диапазоне от 5 до 15).