**Задача 2-1-Максимальный блок**

Описана структура данных

typedef struct list {

void \*address;

size\_t size;

char comment[64];

struct list \*next;

} list;

Требуется реализовать только одну функцию, которая в данном списке находит адрес блока памяти занимающий больше всего места. Адрес хранится в поле address, поле size - соответствующий размер данного блока. Если список пустой, то функция должна возвращать NULL. Если есть несколько таких блоков, то вернуть адрес любого из них. Прототип функции:

void \* findMaxBlock(list \*head)

Пример №1

Данные на входе:



**Задача 2-2-Всего памяти**

Описана структура данных для хранения информации об использованной памяти:

typedef struct list {

void \*address;

size\_t size;

char comment[64];

struct list \*next;

} list;

Требуется реализовать только одну функцию, которая анализирует данный список и возвращает сколько всего памяти используется. Адрес хранится в поле address, поле size - соответствующий размер данного блока. Если список пустой, то функция должна возвращать NULL. Прототип функции:

size\_t totalMemoryUsage(list \*head)

Загрузите только текст данной функции

Пример №1

Данные на входе:



Данные на выходе: 140