

**Демонстрационный вариант заданий для практической части
предпрофессионального экзамена в рамках проекта
«Инженерный класс в московской школе»
на площадке Национального исследовательского университета
«Высшая школа экономики»**

Направление практической части: Программирование

Направление подготовки: Информационные технологии

Программа НИУ ВШЭ: Программирование космических систем на языке C

Тематическое содержание:

Практическая часть содержит одну задачу по программированию на языке C. Экзаменуемым необходимо написать программу для снятия показаний с одного из датчиков обучающего конструктора “Orbcraft” и обработки этих показаний в соответствии с заданием. Для решения задачи школьникам необходимо обладать навыками обработки одномерных и двумерных массивов, создания и использования процедур и функций. Для работы с датчиками должны быть использованы функции библиотеки “libschat.h”, предназначенной для работы с датчиками конструктора “Orbcraft”. Описание библиотеки будет выдано на экзамене в качестве раздаточного материала.

Для успешного решения практической части экзамена обучающиеся должны обладать компетенциями по следующим темам:

Тема 1. Одномерные и двумерные массивы:

- Объявление массивов.
- Заполнение элементов массива с клавиатуры и случайными значениями.
- Циклы для работы с одномерными и двумерными массивами.
- Обращение к диагоналям двумерного массива.
- Алгоритмы сортировки: метод установки, метод пузырька, быстрая сортировка.
- Вывод элементов на экран.

Тема 2. Процедуры и функции:

- Объявление процедуры/функции.
- Вызов процедуры/функции из основной программы.
- Передача параметров в процедуры/функции по значению, по указателю, по адресу.

Тема 3. Работа с функциями библиотеки “libschat.h”:

- Функции включения/выключения солнечных датчиков, датчика угловой скорости, магнитометра.
- Функции для снятия показаний с солнечных датчиков, датчика угловой скорости, магнитометра.
- Функции включения/выключения, изменения угловой скорости и снятия показаний для маховика.

Задание:

Написать программу на языке C, решающую следующую задачу.

Выполнить снятие N ($10 < N < 1000$) трехмерных векторов магнитного поля при помощи магнитометра с интервалом в 1 секунду. Показания записываются в двумерный массив: три значения в каждой строке соответственно равны показаниям, снятым по оси Ox , Oy и Oz за одно измерение. Необходимо найти вектор магнитного поля максимальной длины, вывести на экран его длину, координаты и момент времени после начала отсчёта, когда был зафиксирован этот вектор.

Количество измерений N вводится с клавиатуры. Для снятия показаний необходимо использовать функции включения используемого датчика, снятия показаний и выключения из библиотеки "libschat.h" конструктора "Orbicraft" (описание функций находится в раздаточном материале).

Раздаточный материал.

Описание функций библиотеки "libschat.h"

Для работы со всеми устройствами используется номер 1.

Для реализации задержки программы по времени на t секунд используется функция `Sleep(t)`.

Стандартные коды, возвращаемые функциями библиотеки:

```
#define LSS_OK 0
#define LSS_ERROR 1
#define LSS_BREAK 2
```

Включение датчика угловой скорости (ДУС) номер *num* :

```
int hyro_turn_on(uint16_t num);
```

Выключение ДУС номер *num*:

```
int hyro_turn_off(uint16_t num);
```

Снятие показаний с ДУС номер *num*:

```
int hyro_request_raw(uint16_t num, int16_t *pRAW_dataX, int16_t
*pRAW_dataY, int16_t *pRAW_dataZ);
```

Возвращает LSS_OK и записывает текущие показания ДУС с номером *num* в переменные `pRAW_dataX` - `pRAW_dataZ`

В случае ошибок на шине возвращает LSS_ERROR

В случае отсутствия питания на ДУС возвращает LSS_BREAK

Включение магнитометра номер *num*:

```
int magnetometer_turn_on(uint16_t num);
```

Выключение магнитометра номер *num*:

```
int magnetometer_turn_off(uint16_t num);
```

Снятие показаний с магнитометра номер *num*:

```
int magnetometer_request_raw(uint16_t num, int16_t *pRAW_dataX, int16_t
*pRAW_dataY, int16_t *pRAW_dataZ);
```

Возвращает LSS_OK и записывает текущие показания магнитометра с номером num в переменные pRAW_dataX - pRAW_dataZ

В случае ошибок на шине возвращает LSS_ERROR

В случае отсутствия питания на магнитометре возвращает LSS_BREAK

Включение солнечного датчика номер num:

```
int sun_sensor_turn_on(uint16_t num);
```

Выключение солнечного датчика номер num:

```
int sun_sensor_turn_off(uint16_t num);
```

Снятие показаний с солнечного датчика номер num:

```
int sun_sensor_request_raw(uint16_t num, uint16_t *pRAW_data1,  
uint16_t *pRAW_data2);
```

Возвращает LSS_OK и записывает текущие показания сенсоров солнечного датчика с номером num в переменные pRAW_data1 и pRAW_data2.

В случае ошибок на шине возвращает LSS_ERROR

В случае отсутствия питания на солнечном датчике возвращает LSS_BREAK

Критерии оценивания:

№ п/п	Критерии оценивания	Количество баллов
1	Подключены все необходимые библиотеки	3
2	Правильно выбраны функции библиотеки "libschat.h" (включение/выключение датчиков, снятие показаний с датчиков)	12
3	Верная передача параметров в функции библиотек	5
4	Осуществление ввода значений с клавиатуры и вывода результатов на экран	5
5	Анализ существования результата (вывод сообщений о возможности/невозможности) достижения результата)	5
6	Обработка массивов (сортировка, поиск минимума/максимума и т.д.)	10
7	Правильность алгоритма решения задачи	20

№ п/п	Штрафные баллы	Количество баллов
1	Синтаксическая ошибка (пропущенные знаки ";", "{}", "()", ошибки в операторах и т.д.)	0,5 балла за каждую ошибку
2	Отсутствие комментариев	1
3	Отсутствие отступов	1-5
4	Работа полностью или частично заимствована	60