1.Описание программных функций:

Инициализация данных – создать структуру данных, в которой абитуриенты разнесены по специальност~~и(-~~ям)

Ввод данных – запросить у пользователя название специальности, по которо~~му(-~~й) он хочет получить информацию об абитуриент~~ов (~~-ах)

Поиск абитуриентов – проверить, существует ли введённый жанр??? в программе. (кроме проверки существования требуется, собственно, информация об абитуриентах данной специальности. Дополнить)

Расчёт среднего балла – для каждого абитуриента найти средний балл по аттестату ~~который выше остальных~~

Сортировка абитуриентов – отсортировать абитуриентов (всех?) по среднему количеству оценок??? в порядке убывания.

Вывод результатов – вывести на экран список абитуриентов (всех?) по убыванию среднего балла

2. Описание структуры информации:

Список всех абитуриентов - var ApplicantLibrary = new Dictionary<string, List<Name of the specialty>> - названия типов в C# указывается неразрывно

Расшифровать, что это за структура Dictionary, за что отвечает string, что такое List< Name of the specialty>

Список абитуриентов по среднему баллу - – classApplicant, характеризуется названием string Title, ~~количеством абитуриентов List<int> Estimation~~ (не требуется согласно заданию). Список не может характеризоваться несколькими атрибутами, он хранит конкретные типы. Указать тип, отвечающий за представление информации об абитуриенте и тип, отвечающий за представление всей совокупности информации об абитуриентах.

~~3. Инициализация данных:~~

~~var ApplicantLibrary = new Dictionary<string, List<Name of the specialty>>~~

~~Ввод данных:~~

~~Console.Write(“Введите средний балл: ”);~~

~~string estimation = Console.ReadLine();~~

~~Поиск абитуриентов~~

~~if (ApplicantLibrary.ContainsKey(estimation))~~

Это должно относиться к 4 пункту ЛР.