ПРИЛОЖЕНИЕ 1

МИНОБРНАУКИ РОССИИ

федеральное государственное бюджетное  
образовательное учреждение высшего образования

«ЧЕРЕПОВЕЦКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Институт информационных технологий

наименование института (факультета)

Математического и программного обеспечения ЭВМ

наименование кафедры

Объектно-ориентированное программирование

наименование дисциплины в соответствии с учебным планом

УТВЕРЖДАЮ

Зав. Кафедрой МПО ЭВМ,

д. т.н., профессор Ершов Е.В.

« » 20 г.

Программирование на языке высокого уровня

Техническое задание на курсовой работу

Листов 6

Руководитель

Исполнитель

студент

Ершов Евгений Михайлович

ФИО преподавателя

1ПИб-01-21оп

группа

Лапин Владислав Александрович

Фамилия, имя, отчество

2020 год

1. Введение

Цель курсовой работы: разработать программу на языке C++ с использованием механизмов объектно-ориентированного программирования. Предметной областью является воздушный транспорт.

Данный продукт может быть использован как инструмент для работы с определенными объектами воздушного транспорта посредством пользовательского интерфейса.

1. Основания для разработки

Основанием для разработки является задание на курсовую работу, выданное в рамках учебного плана на кафедре Математического и программного обеспечения ЭВМ по дисциплине «Объектно-ориентированное программирование».

1. Назначение разработки

Основная задача данной курсовой работы – получение новых практических и теоретических знаний, а также закрепление и применение на практике материала, полученного в ходе изучения дисциплины «Объектно-ориентированное программирование».

Программа может быть интересна тем, кто изучает или интересуется программированием и воздушным транспортом. Данный продукт предоставляет возможность работать с объектами класса через пользовательский интерфейс.

1. Требования к программе
2. Требования к функциональным характеристикам

Общие требования:

* корневым классом в иерархии типов должен быть класс-интерфейс;
* взаимодействие иерархии типов должно быть организованно через указатели на интерфейсы и виртуальные функции (отложенные методы);
* должен быть создан обобщенный контейнерный класс (базовый) и от него, используя закрытое наследование, производный класс – шаблон для хранения указателей на абстрактный базовый класс-интерфейс (статический вектор).

Программа должна:

* предоставлять возможность создания записи об объекте с указанием параметров, а также её удаление;
* производить сортировку и фильтрацию записей по определенным параметрам;
* обеспечивать корректный ввод данных с клавиатуры, а также вывод на дисплей;
* обеспечивать сохранение данных в файл и загрузку из него.

1. Требования к надежности

Для хранения объектов каждого производного класса необходимо использовать очередь.

В программе должна быть реализована обработка следующих исключительных ситуаций:

* ввод недопустимого значения (несоответствующий формат; значение, выходящее из допустимого диапазона);
* выход за границы статического вектора;

После обработки исключительной ситуации должно выводиться сообщение об ошибке с указанием рода/типа ошибки.

1. Условия эксплуатации

Данная программа предназначена для работы на одной компьютере одним пользователем.

1. Требования к составу и параметрам технических средств

Минимальные системные требования:

* процессор с тактовой частотой 1800 MHz или более мощный;
* оперативная память 512 Мб или больше;
* видеокарта с памятью 256 Мб или больше;
* свободное место на жёстком диске от 150 МБ;
* архитектура с разрядностью 32 бит или 64 бит;
* наличие монитора, клавиатуры и мыши.

1. Требования к информационной и программной совместимости

Код программы реализуется на языке C++ в среде разработки Visual Studio 2019. Данный продукт может использоваться в операционных системах Windows 7/8/10 при наличии установленного программного обеспечения.

1. Требования к маркировке и упаковке

Требования не предъявляются.

1. Требования к транспортированию и хранению

Требования не предъявляются.

1. Требования к программной документации

Состав программной документации должен включать в себя:

* расчетно-пояснительную записку;
* техническое задание;
* руководство пользователя;
* текст программы.

1. Технико-экономические показатели

Требования не предъявляются.

1. Стадии и этапы разработки

В таблице П1.1 содержится информация обо всех этапах и стадиях разработки программного продукта.

Таблица П1.1

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование этапа разработки ПО | Сроки разработки | Результат выполнения | Отметка о выполнении |
| Разработка технического задания | 21 - 23 марта | Составлено техническое задание |  |
| Анализ предметной области | 23 - 28 марта | Поставлена цель работы, изучено задание и проведен анализ предметной области |  |
| Создание иерархии классов | 28 марта  –  10 апреля | Создана иерархия классов.  Каждый класс спроектирован |  |
| Проектирование интерфейса | 10 – 12 апреля | Разработан макет, готовы формы для интерфейса |  |
| Проектирование программы | 12 – 25 апреля | Разработка программы на языке C++ |  |
| Тестирование программы | 12 - 25 апреля | Исправление обнаруженных ошибок |  |
| Оформление РПЗ | 25-30 апреля | Оформление итоговой документации |  |

1. Порядок контроля и приемки

В таблице П1.2 отражена информация о прядке контроля и приемки.

Таблица П1.2

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование контрольного этапа выполнения курсовой работы | Сроки контроля | Результат выполнения | Отметка о приемке результата контрольного этапа |
| Согласование технического задания | До 2 апреля | Техническое задание согласовано |  |
| Согласование иерархии родственных классов | До 12 апреля | Иерархия родственных классов согласована |  |
| Тестирование программы и обработка исключительных ситуаций | До 1 мая | Программа протестирована, ошибки исправлены, созданы обработчики исключений |  |
| Предоставление программы | 1 мая | Программа проверена преподавателем |  |
| Проверка расчетно-пояснительной записки | До 15 | Проверка расчетно-пояснительной записки |  |
| Защита курсовой работы |  | Защита работы |  |