**Примерный перечень тем курсовых работ по дисциплине *программирование***

|  |  |  |
| --- | --- | --- |
| Категория | Тема | Примечание |
| Информационные и управляющие системы | 1. автоматизированная обучающая система; 2. система дистанционного тестирования; 3. генератор расписания занятий учебного заведения; | Рекомендуется предусмотреть возможность распределенного использования приложения.  В качестве альтернативы для 2,3 вариантов возможна доработка существующей системы – добавление поддержки нового типа вопросов, в котором ответом будет являться программный код. |
| Утилиты, прикладное и системное ПО | 1. разработка файлового менеджера (аналог Total Commander, FAR и т.п.); 2. создать утилиту *Pinger* – мониторинг состояние доступности узлов сети. выводит таблицу доступности; 3. редактор электрических схем (или что-то подобное); 4. написание out-of-process COM/DCOM (или используя аналогичную технологию) сервера, предоставляющего некоторый сервис; 5. плагин/дополнение к существующему ПО; |  |
| Разработка игры | 1. игра "морской бой"; 2. игра в домино; 3. игра в шашки; 4. игра в шахматы; 5. генератор кроссвордов/сканвордов; 6. Разработка игры на Unity (или аналогичном движке) | В играх на 2х и более человек, предусмотреть возможность участия компьютера, реализовать несколько стратегий его поведения (эвристические, мини-макс и другие, в зависимости от задачи)  В вариантах 9, 13 - настройка размеров игрового поля |
| Обработка текста, синтаксический анализ | 1. поиск плагиата в исходных кодах на C/C++; 2. синтаксический анализатор предложений русского языка; 3. программа-переводчик с одного языка на другой; 4. перевод арифметических выражений (АВ) в ПОЛИЗ. Вычисление значения произвольного АВ; 5. построение 2,3 - мерных графиков функций на основе введенного выражения; |  |
| Математические методы в программировании | 1. поиск решения одной из NP полных задач (задача коммивояжера, календарного планирования, упаковки и т.д.); 2. нейронные сети. разработка модели многослойной нейронной сети и ее обучение; 3. математическое моделирование поведения какого-либо объекта или системы; | Решить задачу несколькими методами, сравнить их эффективность |
| Другое | 1. мобильное приложение; 2. веб-приложение; 3. вариант по предложению студента. | Тема согласовывается с преподавателем |

***P.S. Здесь представлен только список тем. Функционал (что должна делать программа, какие задачи решать в рамках этой темы и т.п.) обговаривается отдельно.***

**Состав курсовой работы:**

Курсовая работа должна включать: титульный лист, содержание, введение, основную часть, заключение, список использованных источников, приложения.

Объем курсовой работы составляет 25-30 страниц машинописного текста, набранных на компьютере, не считая приложений.

Основная часть должна включать в себя:

1. постановку задачи;
2. описание разработанной системы, включая иерархию классов, их описание, взаимосвязь компонентов системы и т.д.;
3. требования к аппаратному и программному обеспечению;
4. примеры работы программы.

Курсовая работа должна быть оформлена в соответствии с ГОСТ 7.32-2001 «Отчет о научно-исследовательской работе. Структура и правила оформления».

Отдельно необходимо предоставить:

1. исходные тексты программы;
2. скомпилированную release версию проекта;
3. необходимые для запуска программы фреймворки, библиотеки, компоненты, и т.п. (в крайнем случае предоставление этого набора в виде образа виртуальной машины).

**Критерии оценки:**

корректность постановки и решения задачи;

знание и использование принципов ООП;

работоспособность программы;

ответы на вопросы при защите;

содержание и оформление отчета.

|  |  |
| --- | --- |
| **+** | **-** |
| Групповая работа с использованием системы контроля версий | нарушение контрольных сроков |
| возможность редактирования, отмены и возврата действий (более 2-х) | отход от первоначального, согласованного задания |
| наличие контекстной справки | использование среды разработки старше 5 лет |
| возможность генерации отчетов (если позволяет задание) в форматах excel, word, pdf или аналогичных. | параметры программы жестко зашиты в программу |
| использование unit-тестирования в процессе разработки | хранение, передача паролей в открытом виде |
| использование статических анализаторов кода | наличие грубых ошибок в программном коде, в том числе, отсутствие процедур проверки целостности, непротиворечивости данных) |
| соблюдение паттернов проектирования (в отчете указать, где и какие использованы) | хранение данных в бинарных или текстовых файлах, там, где целесообразнее использовать БД. |
| Наличие руководства пользователя |  |

**Требования к языкам программирования, используемым технологиям:**

Без ограничений. Единственное условие – при использовании редкоиспользуемых сред/языков, вместе с исходными кодами программы нужно предоставить образ виртуальной машины с настроенной средой для запуска проекта.