



Практические задания (кейсы)

1 Название проекта: Работа с MongoDB.

Цель проекта: Проверка умения работать с базами данных.

2 Формат применения: тестовое задание.

4 Задание: написать функцию, которая получает N случайных, не повторяющихся документов из коллекции.

Будем бонусом: рассмотреть разные методы решения данной задачи, выделить достоинства и недостатки каждого метода.

Примечание:

- Если в коллекции содержится M документов, при этом $M < N$, функция должна вернуть M документов.
- После написания функции, требуется написать несколько тестов, которые будут демонстрировать качество и скорость работы функции.

5 Тестовое задание должно быть выполнено на языке TypeScript

6 Возрастная категория: школьник, учащийся вуза.



Название проекта: Основы React.

1. Цель проекта: Проверка знаний основы работы с React.js.
2. Формат применения: тестовое задание.
3. Задание: создать одностраничное React приложение, в котором необходимо:
 - Создать компонент, имеющий состояние: число, функции INC, DEC, RND, RESET, которые будут добавлять к текущему значению +1 или -1, прибавлять случайное значение или делать сброс в исходное состояние соответственно.
 - Добавить в первый компонент запрос на <https://www.random.org/integers/?num=1&min=-50&max=50&col=1&base=10&format=plain&rnd=new>, чтобы при каждой перезагрузке страницы вместо 0 ставилось случайное значение.
 - Создать компонент конвертера валют со следующими элементами: select-ы с выбором валют в которую и из которой будем переводить сумму, поле для ввода суммы, просмотр результата конвертации

Дополнительные задания:

- Вынести логику работы функций INC, DEC, RND, RESET в отдельный собственный хук (функцию) и использовать ее для первого компонента.
- Брать данные текущего курса с открытых API, а не вводить вручную (второй компонент)

Тестовое задание должно быть выполнено на языке TypeScript

Возрастная категория: школьник, учащийся вуза.



1. **Название проекта: Приложение-задачник.**

Цель проекта: Проверка знаний основы работы с React.js.

2. **Формат применения: тестовое задание.**

Возрастная категория: школьник, учащийся вуза.

3. **Задание: необходимо создать одностраничное приложение-задачник (ToDo list) на React с использованием локального хранилища (localStorage).**

Список задач должен сохраняться при обновлении страницы.

Задачи в пользовательском интерфейсе состоят из:

- **текст задачи**
- **статус выполнения**

Тестовое задание должно быть выполнено на языке TypeScript



Название проекта: Файловое хранилище.

- 1 Цель проекта: Проверка умения работать с базами данных и знания основы серверной разработки.**
- 2 Формат применения: тестовое задание.**
- 3 Задание: Напишите сервер, который позволяет создавать хранилища файлов, например media, books, executable files и т.д.**

Примечание:

- Настройка хранилищ должна осуществляться из кода.**
- Нужно реализовать авторизацию.**
- Каждое хранилище должно иметь свои параметры, например максимальный вес файла, допустимые типы и т.д.**
- API сервера должно содержать роуты для взаимодействия с хранилищами - загрузка файла, скачивание файла, просмотр информации о файле или хранилище в целом и т.д.**
- Можно использовать любую базу данных.**

Тестовое задание должно быть выполнено на языке TypeScript

Возрастная категория: учащийся вуза.



Название проекта: Приложение-задачник (fullstack).

- 1. Цель проекта: Проверка знаний основы работы с React.js, Redux, Node.js.**
- 2. Формат применения: тестовое задание.**
- 3. Задание: необходимо создать приложение-задачник (ToDo list). Backend на Node.js (Express), frontend на React с использованием центрального хранилища (redux, mobx или context provider). База данных - любая.**

Задачи в пользовательском интерфейсе состоят из:

- **имя пользователя**
- **email**
- **текст задачи**

Стартовая страница - список задач с возможностью сортировки по имени пользователя, email и статусу.

- **Вывод задач нужно сделать страницами по 3 штуки (с пагинацией).**
- **Видеть список задач и создавать новые может любой посетитель без авторизации**

Сделайте вход для администратора (логин "admin", пароль "123").

- **Администратор имеет возможность редактировать текст задачи и поставить галочку о выполнении.**
- **Выполненные задачи в общем списке выводятся с соответствующей отметкой**

Тестовое задание должно быть выполнено на языке TypeScript

Длительность выполнения задания: 1 неделя.

Возрастная категория: учащийся вуза.



Название проекта – Сервис получения и отображения прогноза погоды по текущей гео-позиции пользователя.

1. Цель проекта – научить участников созданию простых веб приложений.
2. Формат применения кейс-задания - учащийся вуза.
3. Исходные данные, условия, ограничения

Сценарий:

Пользователь заходит на web-страницу сервиса, на котором отображается информация:

- город, в котором находится пользователь
- текущая дата
- минимальная, максимальная температура на день
- скорость и направление ветра на текущий день
- иконка для соответствующего состояния погоды на текущий день
- прогноз на ближайшие 5 дней с аналогичной информацией

Требования к реализации:

Дизайн на усмотрения разработчика, но пользоваться сервисом должно быть приятно.

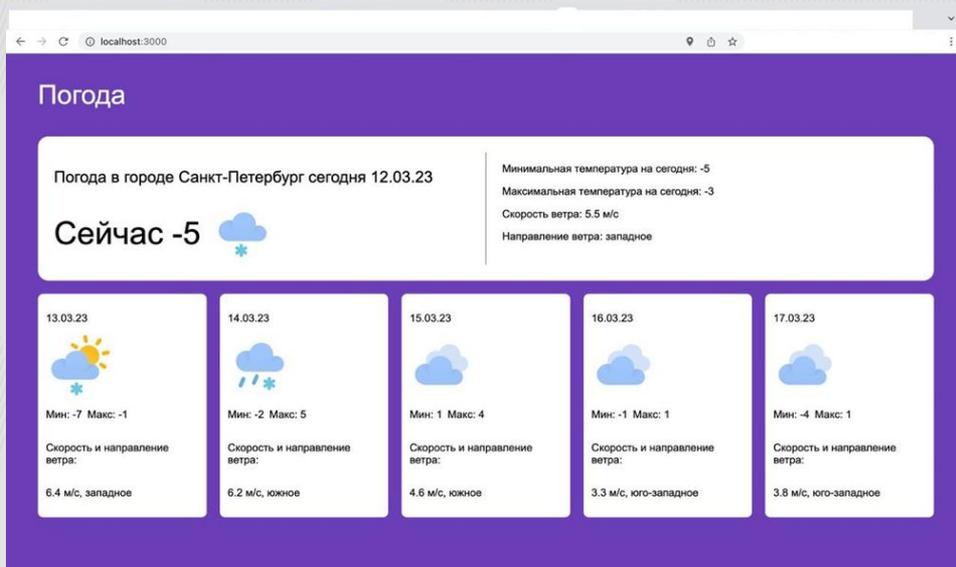
Язык программирования: приветствуется TypeScript, но можно и JavaScript.

Стек: на фронте – React, на бэке Express или NestJs

Данные погоды можно получить по API Яндекс Погода по бесплатному тарифу “Тестовый” (<https://yandex.ru/dev/weather/doc/dg/concepts/about.html>)

4. Требования к конечному результату, программному обеспечению, используемым материалам, длительности выполнения задания.

Возможный вид web-страницы:



Используемые материалы: интернет (не ограничено)



Аппроксимация двухфакторных наборов данных степенными функциями / полиномами

1. Цель проекта: найти оптимальные варианты аппроксимирующих функций для двухфакторных наборов данных.
2. Исходные данные, условия, ограничения:

Есть прибор, который измеряет давление и температуру жидкости

в гидравлической системе с помощью резонансных чувствительных элементов. Выходным параметром чувствительных элементов является частота их колебаний, которая зависит от давления и температуры жидкости

в гидравлической системе.

В прилагаемом документе Excel «Наборы данных для проектного задания 357-1 полином» на вкладках представлены три набора данных.

Каждый набор данных представляет из себя:

столбец «А» – значение давления жидкости в гидравлической системе P_d ;

столбец «В» – значение частоты выходного сигнала чувствительного элемента измерения давления f_p ;

столбец «С» – значение частоты выходного сигнала чувствительного элемента измерения температуры f_t .

Например, в строке 3 на вкладке «Набор данных 1» записано значение давления $P = 300$ условных единиц, которому соответствует частота колебаний чувствительного элемента измерения давления $f_p = 45344.33$ условных единиц, и при этом значение частоты колебаний чувствительного элемента измерения температуры составляет $f_t = 32824.294$ условных единиц.



Необходимо найти 2 ÷ 3 варианта степенной функцию / полинома

$P = F(f_p, f_t)$, которые достаточно точно описывали бы представленные данные.

Критерий достаточной точности – модуль отклонения рассчитанных

с помощью функции значений давления P_p от представленных значений давления P_d в наборах данных не должно превышать 10 условных единиц

$$\Delta = |P_p - P_d| < 10.$$

Степень функции не должна превышать 5.

3. Требования к конечному результату:

Расчеты давления P_p с помощью функции и величины его отклонения Δ от заданных значений должны быть представлены в Excel. Приветствуется не только табличная, но и графическая форма представления результатов.

Возрастная категория: Студенты ВУЗов.



Программное обеспечение для микроконтроллера. Частотомер.

1. Цель проекта: разработать программное обеспечение для микроконтроллера, измеряющее частоту входных импульсов в диапазоне 25кГц-55кГц.
2. Исходные данные, условия, ограничения:

На вход микроконтроллера подаются прямоугольные импульсы (меандр) с частотой следования 30кГц – 50кГц. Кратковременная нестабильность частоты следования импульсов не хуже 10⁻⁶.

Необходимо разработать программное обеспечение, которое определяет частоту входных импульсов и передает информацию

по универсальному асинхронному приемо-передатчику (UART) через микросхему интерфейса RS-232 или RS-485.

Тип микроконтроллера – предпочтительно из серии STM32.

Алгоритм вычисления, а так же значение опорной частоты для подсчета входных импульсов выбирает разработчик.

Программа должна производить измерение частоты,

как по запросу, так и в автоматическом режиме. Интервал времени между передаваемыми сообщениями с результатами измерений должен задаваться по команде в диапазоне от 1 до 99 сек.

Протокол передачи данных – содержание передаваемых и принимаемых сообщений определяет разработчик.



3. Требования к конечному результату:

- время измерения не должно превышать 150мс;
- нестабильность измерения частоты не должна превышать пределов $\pm 0,1$ Гц при постоянном значении температуры окружающей среды;
- программа должна выводить значение измеренной частоты до 3-го знака после запятой. Пример: 35897,135 Hz .

Возрастная категория: Студенты ВУЗов.



Радиолиния передачи данных от беспилотного воздушного судна

1. Цель проекта: выбрать оптимальный состав радиоканала для передачи данных с беспилотного воздушного судна, расчётным путём определить и обосновать параметры радиоканала
2. Исходные данные, условия, ограничения:

Задание 1

Предложить состав бортового оборудования беспилотного воздушного судна (далее – БВС) и оборудования наземной станции управления (далее – НСУ), обеспечивающих получение на расстоянии до 100 км данных с цветной видеокамеры БВС с качеством Full HD. Определить и обосновать параметры радиоканала. Составить профиль оптимальной высоты полёта БВС с учётом обеспечения максимальной скрытности БВС от средств визуального и радиолокационного обнаружения с поверхности земли.

Задание 2

Определить оптимальный вид модуляции и максимальную скорость передачи данных без использования помехоустойчивого кодирования в радиоканале с БВС на расстоянии 50 и 200 км от НСУ. Для расчёта использовать параметры антенн и радиопередающих средств, определённых при выполнении задания 1. Ширина полосы излучения не должна превышать 5 МГц. Частота возникновения битовых ошибок в радиоканале (далее - BER) не должна превышать 10^{-5} .

Задание 3

Предложить и обосновать варианты повышения помехозащищённости радиоканала с БВС при тех же значениях ширины полосы излучения (далее – ШПИ) и BER, что и в задании 2. Оценить энергетический выигрыш (сделать сравнительный анализ) предложенных вариантов.

2 Требования к конечному результату:

Предоставить:

- структурную схему радиоканала с обоснованием параметров (видеокодек, скорость передачи, параметры антенн, приёмопередающих радиосредств БВС и НСУ, потери мощности сигнала на трассе);
- расчёт энергетического баланса радиоканала (отношение энергии бита к спектральной плотности мощности шума - E_b/N_0);
- график оптимальной высоты полёта БВС с указанием отношения E_b/N_0 в контрольных точках на расстоянии 25 км, 50 км, 75 км и 100 км от НСУ;
- графическое обоснование выбора вида модуляции и расчёт максимальной скорости передачи данных на расстоянии 50 км и 200 км от НСУ при заданных ШПИ и BER;
- обоснование вариантов помехозащиты с их сравнительным анализом.



Информационная безопасность на объекте информатизации

1. Цель проекта: Разработка проектно-архитектурного решения для системы менеджмента инцидентов информационной безопасности комплекса средств автоматизации информационных процессов.
2. Исходные данные, условия, ограничения:

Объект информатизации представляет собой комплекс средств автоматизации, связанных между собой в локальную (не связанную с глобальной) сеть, предназначенных для решения специальных задач и функционирующих под управлением ОС Astra Linux.

Задание 1

Описать модель угроз информационной безопасности:

- классификация угроз и их источников;
- характеры угроз, их воздействие;
- факторы (причины уязвимости);
- характер и последствия ущерба.

Оценить исходные данные и определить перечень возможных уязвимостей.

Задание 2

Разработать проектно-архитектурное решение для системы управления безопасностью КСА в части, касающейся информационно-технического и программно-аппаратного обеспечения.

Решение должно обеспечивать:

- авторизацию пользователей;
- разграничение доступа;
- защиту информации в процессе хранения;
- контроль над отчуждаемой информацией;
- инвентаризацию сети;
- фиксацию и контроль целостности защищаемой информации;
- аудит уязвимостей и безопасности объекта информатизации;
- аудит защищённости баз данных;
- аудит серверов приложений;
- аудит сетевого оборудования.

Задание 3

Предложить варианты информационно-технических и программно-аппаратных средств защиты информации от отечественных производителей и разработчиков. Выполнить их сравнительный анализ и обосновать выбор.

3. Требования к конечному результату:

В ходе проделанной работы должен быть выполнен анализ угроз информационной безопасности на защищаемом объекте, разработана структурная схема КСА с интегрированным комплексом систем защиты информации, определены программно-аппаратные и технические средства обеспечения защиты информации.



Потокобезопасная очередь

1. Цель проекта: получение информации о навыках работы кандидата с многопоточными приложениями.
2. Исходные данные, условия, ограничения:
разработать шаблонный класс, представляющий потокобезопасную очередь в котором необходимо:
 - реализовать метод, добавляющий элемент в конец очереди;
 - реализовать метод, удаляющий элемент из начала очереди;
 - реализовать метод, возвращающий количество элементов в очереди;
 - реализовать метод, возвращающий информации о пустоте очереди (true – очередь пуста, false – не пуста);
3. Требования к конечному результату:
 - время выполнения: 5 дней;
 - в классе могут быть использованы стандартные контейнеры и примитивы синхронизации;
 - допустимо использовать стандарты языка C++ 11 или C++ 14.
5. Возрастная категория: студенты высших учебных заведений.



Пул потоков

Цель проекта: получения информации о навыках кандидата по разработке многопоточных приложений.

Исходные данные, условия, ограничения:

разработать класс, реализующий работу пула потоков, в котором необходимо:

реализовать конструктор, в который передается количество рабочих потоков;

метод, запускающий работу потоков (init);

метод, ожидающий выполнения всех рабочих потоков и закрывающий работу пула потоков;

метод, отправляющий функцию для асинхронного выполнения пулом потоков;

Требования к конечному результату:

для реализации пула потоков желательно использовать потокобезопасную очередь, реализованную ранее;

в классе должны быть удалены конструкторы копирования, перемещения, а также перемещающий и копирующий операторы присваивания;

допустимо использовать стандарты языка C++ 11 или C++ 14.

Возрастная категория: студенты высших учебных заведений.



Задача 1 «Программирование на Python»

Цель проектного задания:

Проверка базовых знаний языка программирования Python.

Формат применения кейс-задания:

Задание для отбора студентов к прохождению практики.

Исходные данные, постановка задачи и требования к результату:

На языке программирования Python необходимо написать следующие программы:

1) Программа, которая находит и выводит все пифагоровы тройки ($x^2+y^2=z^2$) для $z \leq 200$, а также выводит их количество.

2) Программа, которая вычисляет и выводит N-ное простое число (например, сотое — 541). Номер простого числа должен вводиться в программу через клавиатуру.

3) Программа, которая создаёт массив, столбцы которого идентичны заданным массивам. Элементы массива в каждом его столбце должны быть отсортированы в порядке возрастания значения (сверху вниз). Значения всех элементов, расположенных на диагоналях массива, должны быть заменены на цифру 5. После чего данный массив объединяется по третьему измерению с массивом такого же размера, но заполненного случайными числами

от -10 до 0. Первые строчки кода:

```
- colum_1 = np.array([0, 1, 3, 1, 0])
```

```
- colum_2 = np.array([1, 3, 1, 4, 1])
```

```
- colum_3 = np.array([4, 1, 3, 5, 4])
```

```
- colum_4 = np.array([4, 8, 4, 8, 5])
```

```
- colum_5 = np.array([7, 1, 3, 1, 0])
```

Программа, которая принимает на вход файл, содержащий погодные данные города Санкт-Петербург за 2022 г. с января по апрель, и отстраивает график средней (между днем и вечером) температуры, а также гистограмму «облачности». В качестве текстового файла используйте файл «gismeteo-spb.txt». Файл представляет собой таблицу, разделенную запятыми, содержащую следующие значения: Месяц, Число, Время, Температура, Явления, Облачность, Ветер, Направление.

Месяц — порядковый номер месяца в 2022 году

Число — число месяца

Время — время замера:

0 — вечер

1 — день

Температура — целочисленное значение температуры в градусах цельсия

Давление — целочисленное значение атмосферного давления в мм ртутного столба

Явления — виды погодных явлений:

0 — нет явлений

1 — дождь

2 — снег

3 — гроза

Облачность — одно из описаний неба:

0 — малооблачно

1 — облачно

2 — пасмурно

3 — ясно

Ветер — скорость ветра в м/с

Направление — направление ветра:

0 — восток

1 — запад

2 — север

3 — северо-восток

4 — северо-запад



5 — штиль (нет ветра)

6 — юг

7 — юго-восток

8 — юго-запад

При создании программы требуется использовать преимущественно библиотеки «numpy» и «matplotlib» для всех операций.

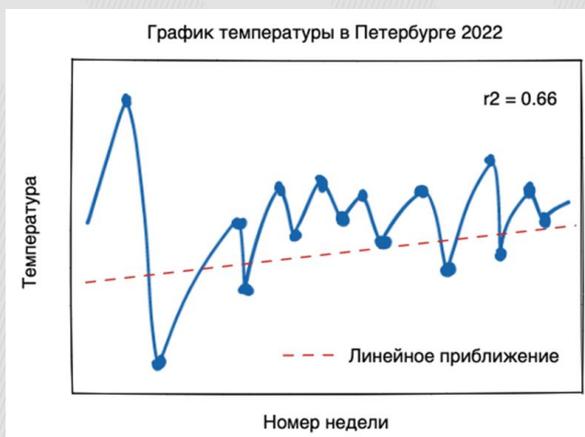
Загрузите данные в память любым удобным способом, например «np.loadtxt» или стандартным «open».

При помощи «matplotlib» отстройте график средней (между днем и вечером) температуры.

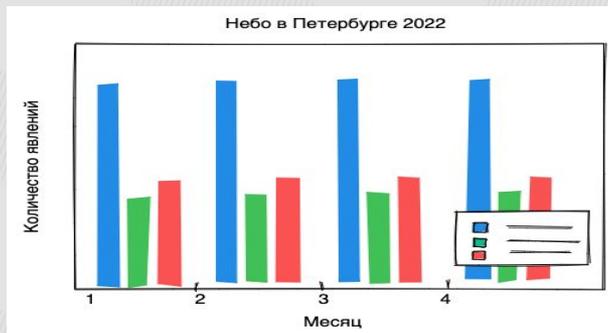
При помощи «numpy» проведите прямую, линейно приближающую получившийся график.

Добавьте подписи осей и заголовок, легенды к линиям на графике и посчитанный коэффициент.

Результат должен выглядеть примерно так:



Постройте гистограмму «облачности», т. е. для каждого месяца отложите количества встречаемости того или иного описания неба. Разукрасьте столбцы каждого явления своим цветом. Ниже представлен пример гистограммы:



Требования к программному обеспечению и срок выполнения проектного задания:
Для выполнения проектного задания необходима операционная система (Желательно Windows или Ubuntu);
один из браузеров: Yandex Browser, Chrome, Chrome Mobile, Firefox, Opera, Safari, Mobile Safari, Edge. Код на языке программирования Python должен быть реализован в среде Jupyter Notebook или с помощью браузера в среде Google Colab.



Задача 2 «Градиентный бустинг против других моделей»

Цель проектного задания:

Проверка знаний о основных техниках предобработки данных и навыков применения их для обучения нескольких моделей: логистической регрессии, дерева решений, случайного леса и модели градиентного бустинга над деревьями решений.

Формат применения кейс-задания:

Задание для отбора студентов к прохождению практики.

Консультант от подразделения по вопросам реализации задания:

Исходные данные:

Для выполнения задания необходимо использовать предоставленный набор данных, который включает в себя информацию о 6000 заявок на гранты, поданных в Мельбурнский университет в период с 2004 по 2008 год.

Полное описание данных можно найти на Kaggle: <https://www.kaggle.com/c/unimelb>.

Постановка задачи и требования к результату:

Суть задания состоит в том, чтобы на основе 38 признаков, связанных с заявкой на грант, предсказать, будет ли она принята. Из датасета необходимо выделить целевую переменную «Grant.Status» и обозначить её за «у».

Другими словами, сделать так, чтобы «Х» обозначал обучающую выборку, а «у» — ответы на ней. Статус гранта «1» представляет собой успешную заявку на грант, а статус «0» — неудачную. Реализованная модель должна предсказывать вероятность успеха для заявки на грант. Для выполнения задания необходимо воспользоваться следующими методами: логистическая регрессия, деревья решений, случайный лес и градиентный бустинг

Первым делом необходимо осуществить предобработку данных, а именно кодирование категориальных признаков, а также удаление или интерпретация пропущенных значений (при наличии того или другого).

Пропущенные значения можно так же интерпретировать, для этого существует несколько способов, они различаются для категориальных и вещественных признаков. Для вещественных признаков: заменить на 0, заменить на среднее. Для категориальных: интерпретировать пропущенное значение, как ещё одну категорию.

После получения набора данных, для которого выполнены ограничения логистической регрессии на входные данные, необходимо обучить на них регрессию, используя имеющийся в библиотеке «sklearn» функционал по подбору гиперпараметров модели

```
optimizer = GridSearchCV(estimator, param_grid)
```

При обучении линейной модели используйте параметр `solver='liblinear'`.

Также необходимо обучить дерево решений, случайный лес и в завершении обучить модель градиентного бустинга. Для метода градиентного бустинга можно использовать как пакет sklearn, так и библиотеку XGBoost или CatBoost.

Требования к программному обеспечению и срок выполнения проектного задания:

Для выполнения проектного задания необходима операционная система (Желательно Windows или Ubuntu); один из браузеров: Yandex Browser, Chrome, Chrome Mobile, Firefox, Opera, Safari, Mobile Safari, Edge. Код на языке программирования Python должен быть реализован в среде Jupyter Notebook или с помощью браузера в среде Google Colab.



Визуализация движения самолета-носителя РЛС:

Нанести на гугл-карту траекторию самолета по записанным показаниям навигационной системы (широта, долгота).

Нанести след диаграммы направленности на поверхность Земли (произвести сечение плоскости конической поверхностью).

Исходные данные и материалы предоставляются НПК РЭС.

Географическая привязка:

Перевести радиолокационное изображение из криволинейных координат в прямоугольные с помощью интерполяции.

Исходные данные и материалы предоставляются НПК РЭС.

Датчик вибрации:

Подключить пьезоэлектрический сенсор к АЦП Ардуино.

Написать код вывода данных по СОМ-порту.

Исходные данные и материалы предоставляются по запросу



Компенсация механических вибраций:

Напечатать на 3-D принтере двойную плоскую пружину. Соединить пружину с линейным двигателем.

Напечатать на 3-D принтере одинарную плоскую пружину. Соединить её с двигателем с другой стороны.

Подобрать груз к пружине так, чтобы компенсировать вибрации конструкции.

Исходные данные и материалы предоставляются предприятием.

Найти коды Баркера:

Написать скрипт в Matlab для поиска и генерации кодов с минимальными боковыми лепестками – кодов Баркера.

Исходные данные и материалы предоставляются НПК РЭС.

Генерация M-последовательности с помощью сдвигового регистра:

Собрать схему генерации M-последовательности на сдвиговом регистре и Ардуино.

Сгенерировать и сохранить две M-последовательности.

Написать скрипт в Matlab для проверки свойств M-последовательностей, провести их исследование.

Исходные данные и материалы предоставляются предприятием по запросу.



Эксперимент с желобом:

Установить в желобе поплавок. Поплавок должен колебаться.

Менять положение дальних стенок желоба, при котором будет наблюдаться стоячая волна.

Оценить длину волны и ее скорость.

Исходные данные и материалы предоставляются предприятием.

Задача фазированных источников волн:

В емкости с водой от одиночного поплавка наблюдается сферическая волна в ближней зоне и плоская волна в дальней.

Устанавливается пять одинаковых поплавков. Если колебания их синфазны, то наблюдается луч. Если колебания не синфазны, то наблюдается поворот луча.

На противоположной стороне емкости устанавливается поплавок. Поплавок колеблется, наблюдаются синфазные колебания пяти поплавков на другой стороне.

Имитируется прием сигнала от точечного источника.

Исходные данные и материалы предоставляются предприятием по запросу.



Задача распространения волн вдоль земной поверхности:

Написать скрипт в Matlab для оценки относительной амплитуды принимаемого сигнала с учетом интерференции для различных положений излучателя и отражателя.

Проиллюстрировать зависимость относительной амплитуды сигнала от высоты приемника и отражателя.

Исходные данные и материалы предоставляются НПК РЭС.

Задача триангуляции:

Описать способ определения положения объекта с помощью измерений дальности до него. Сколько измерений необходимо для определения положения объекта на плоскости и в пространстве?

Исходные данные и материалы предоставляются по запросу



Алгоритм Хаффмана:

Составить кодовое дерево и закодировать свое ФИО.

Написать крипт в Matlab.

Исходные данные и материалы предоставляются НПК РЭС.

Определение частоты повторения букв:

Написать скрипт в Matlab, который на вход принимает любой текстовый файл.

Обработать его и посчитать частоту повторения каждого символа.

Закодировать алгоритмом Фано и сравнить с кодом Морзе.

Исходные данные и материалы предоставляются по запросу



1) Название проекта: датчик парковки

Датчик парковки с автономным источником питания применяется на открытых и крытых парковках, при любых погодных условиях определяет наличие на парковочном месте транспорта и передает данные на базовую станцию.

Цель проекта: разработка Датчика парковки

2) Формат применения кейс-задания: Для отбора студентов к прохождению практики

3) Исходные данные, условия, ограничения: сеть интернет, программы для расчетов и модулирования физических моделей и электрических, САПР.

4) Требования к конечному результату, программному обеспечению, используемым материалам, длительности выполнения задания:

Размер устройства не более 200x200x50мм, устройство должно быть влаго-/пылезащищённым (IP64), иметь автономный срок работы в режиме ожидания не менее 2 месяцев при парковке 50-ти автомобилей в день, механическая прочность должна предполагать возможный многократный наезд колеса автотранспорта. При занятии парковочного места и при его освобождении должен передаваться сигнал на расстояние от 50 метров на базовую станцию. Устройство должно исправно функционировать при любых погодных условиях.

Конечный результат предполагает описание методов достоверного определения парковки автомобиля, описание логики работы устройства, подробный расчёт времени автономной работы, функционирующий макет.



1) Название проекта: 3D моделирование и прототипирование.

Цель проекта: предоставить в упрощённом виде документы, которые разрабатываются или в разработке которых принимает участие коллектив конструкторской лаборатории в рамках эскизного проекта.

2) Формат применения кейс-задания:

Для отбора студентов к прохождению практики

3) Исходные данные, условия, ограничения:

Выбор объекта разработки:

- Выбрать объект разработки – сборочную единицу (см. ГОСТ 2.101 п.5.1.3). Преимущество отдавать изделиям курсовых или дипломных проектов, возможен подбор конструкции из справочной литературы;
- Ознакомиться с этапами разработки изделий (ГОСТ 2.103, ГОСТ Р 15.101, ГОСТ Р 15.301);
- Ознакомиться с комплектом документов, выпускаемым на разных этапах разработки (ГОСТ 2.103, ГОСТ 2.118 - ГОСТ 2.120, ГОСТ Р 15.301).

Справочные материалы:

- а) Справочник конструктора-машиностроителя. Анурьев В.И.
- б) Основы конструирования. Орлов П.И.
- в) Элементы механизмов. Кожевников С.Н.
- г) ГОСТ 2.101, ГОСТ 2.102, ГОСТ 2.103, ГОСТ 2.118 - ГОСТ 2.120;
- е) ГОСТ Р 15.101, ГОСТ Р 15.301.

Написание краткого технического задания:

Ознакомиться с нормативными документами, которые используются в написании ТЗ (ГОСТ Р 2.105, ГОСТ 15.016);

Написать краткое ТЗ на разработку/конструирование выбранного устройства.

Справочные материалы: ГОСТ Р 2.105, ГОСТ 15.016;

Создание электронной модели:

Ознакомиться с ПО Creo Elements Pro;

Создать трёхмерную модель деталей и сборочных единиц изделия с разной степенью детализации.

Справочные материалы: ГОСТ 2.052 - ГОСТ 2.058; Справочная литература по работе в Creo.

Примечание – Возможно использовать любое другое доступное конструкторское ПО.

Проведение расчёта:

При наличии соответствующего пакета провести расчёт на прочность или тепловой расчёт (на выбор).

Подготовить результаты расчёта для оформления пояснительной записки.

Справочные материалы: ГОСТ Р 2.105, ГОСТ Р 2.106, ГОСТ 2.119; ГОСТ 7.32.

Создание конструкторских и схемных документов:

- Выполнить чертёж общего вида (ГОСТ 2.119).
- Выполнить габаритный чертёж (ГОСТ 2.109).
- Выполнить схему деления на составные части (ГОСТ 2.711).
- Выполнить схему принципиальную, соединений или подключений (в случае работы над электронным узлом) (ГОСТ 2.701, ГОСТ 2.702).

Для обозначения выбранного изделия и его документов подобрать номер по классификатору ЕСКД (ГОСТ 2.201, Классификатор ЕСКД). Вместо четырёхбуквенного кода предприятия указать свои инициалы.

Справочные материалы:

- а) ГОСТ 2.104, ГОСТ Р 2.106;
- б) ГОСТ 2.109, ГОСТ 2.119;
- в) ГОСТ 2.201, Классификатор ЕСКД.
- в) ГОСТ 2.701, ГОСТ 2.702, ГОСТ 2.711.

Написание итоговых текстовых документов:

- Написать краткую пояснительную записку по результатам разработки (ГОСТ 2.106, ГОСТ 2.119).
- Составить ведомость эскизного проекта (ГОСТ Р 2.106, ГОСТ 2.119), в которой перечислить выпущенные документы

Справочные материалы: ГОСТ Р 2.105, ГОСТ Р 2.106, ГОСТ 2.119; ГОСТ 7.32.

5) Требования к конечному результату, программному обеспечению, используемым материалам, длительности выполнения задания:

Конечный результат: разработанные документы:

- Краткое техническое задание (ТЗ). (1-3 листа);
- Схема деления структурная (Е1);
- Схему принципиальную, соединений или подключений. (Э3, Э4, Э5) (при необходимости);
- Чертёж общего вида (ВО);
- Габаритный чертёж (ГЧ);
- Краткую пояснительную записку (ПЗ). (2-5 листов);
- Ведомость эскизного проекта;
- 3D модель.

Кусочно-линейная аппроксимация функции 2 аргументов

1. Цель проекта – реализовать программно кусочно линейную аппроксимацию функции 2 аргументов.
Применимость – калибровочная зависимость показаний датчика от температуры и влажности.

Целью исполнителя является

1. знакомство с принципами работы измерительных систем.
2. знакомство с принципами калибровки (порядок выполнения калибровки).
3. знакомство с различными функциями 2 переменных
4. геометрическая интерпретация функции 2 переменных. (поверхность, пересечение поверхностей).
5. аналитическое построение кусочно-линейных функций двух переменных. Самый простой вариант – поверхность первого порядка (плоскость). Т.е. – по 1 функции на каждый аргумент. Задаем коэффициенты – требуется построить зависимость и при задании 2 аргументов получить значение функции. Далее можно усложнять – более 1 функции на каждый аргумент.
6. более сложный вариант – понятие сплайна и освоение сплайн технологии хотя бы на уровне сплайнов первого порядка.
7. подготовка исходных данных для построения функции.
8. более сложный вариант – подготовка плана эксперимента и определение коэффициентов на основе решения системы линейных уравнений.

2. Исходные данные, условия, ограничения

2. Исходные данные, условия, ограничения

В простейшем варианте – задаем 3 коэффициента для построения функции.

2 уровень сложности – задаем калибровочные точки.

3 уровень – необходимо самостоятельно сформировать план эксперимента и получить калибровочные точки (например выбрав из таблицы случайных чисел).

Кусочно линейная аппроксимация функции 2 аргументов.

Функция меняется в пределах от 0 до 1. Аргументы меняются в пределах от 0 до 1.

Требования к конечному результату, программному обеспечению, используемым материалам.

Конечный результат – программа аппроксимации. Программная среда – на выбор участника. Представление результатов – трехмерная поверхность.



Макет датчика внутричерепного давления методом аддитивной печати

1. Цель проекта – рассчитать и реализовать, например, на 3D принтере зонд для измерения внутричерепного давления.

Проект ориентирован на создание элементов системы мониторинга с использованием пассивных беспроводных датчиков. В качестве одного из вариантов – датчики на поверхностных акустических волнах (ПАВ).

Конкретная система мониторинга – внутричерепное давление.

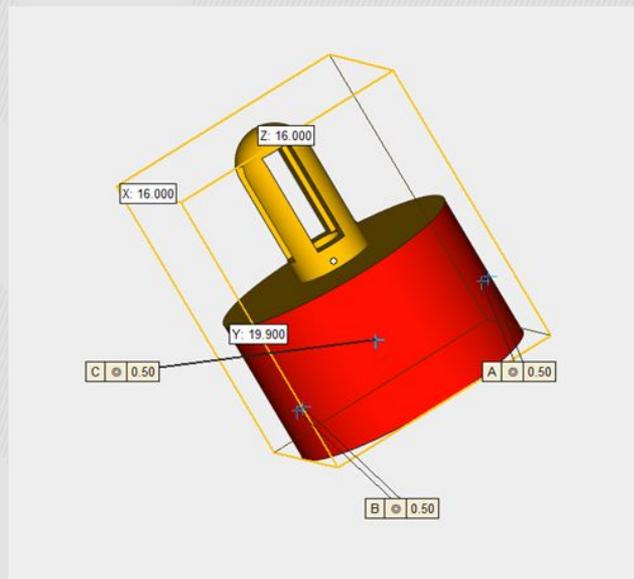
Целью исполнителя является

1. ознакомление с принципами работы системы мониторинга.
2. знакомство с технологиями изготовления изделий малыми партиями.
3. выбор технологии изготовления и соответствующего оборудования.
4. подготовка задания для изготовления.
5. анализ полученных результатов (сравнение результатов между собой и с предложенным эскизом).
6. если есть возможность – оценить прочностные характеристики разных образцов.

2. Исходные данные, условия, ограничения

Эскиз предоставляет заказчик. Приложение 1.

Подобрать технологию, сделать модель и изготовить зонд в соответствии с эскизом



Критерий - минимальная стоимость.

Требования к конечному результату, программному обеспечению, используемым материалам.

Конечный результат – физическая модель зонда. Материал требуется выбрать.



Схема генератора прямоугольных импульсов

1. Цель проекта: выбрать оптимальные схемы генераторов прямоугольных импульсов.
2. Исходные данные, условия, ограничения:

Задание 1

Найти в доступных источниках информации (учебная литература, справочники, ИНТЕРНЕТ) несколько вариантов (более трех) схем электрических принципиальных генераторов с возбуждением от кварцевого низкочастотного резонатора на основной гармонике последовательного резонанса.

На выходе схемы должны формироваться прямоугольные импульсы.

Частота возбуждения основной гармоники должна лежать в диапазоне от 10 до 100 кГц.

Кратковременная нестабильность частоты следования прямоугольных импульсов должна быть не хуже 10^{-6} при постоянном значении температуры окружающей среды и напряжения питания.

Напряжение питания постоянного тока 3,3 В.

Теоретическая часть:

- описать достоинства и недостатки каждой из рассматриваемых схем, влияние изменения температуры и напряжения питания на характеристики выходного сигнала;
- разобраться в способах компенсации влияния температуры окружающей среды.

Практическая часть:

Собрать два, по вашей оценке, лучших варианта схем генераторов с возбуждением от кварцевых резонаторов часового типа.

Задание 2

Разработать схему устройства, которое содержит в себе два генератора

с возбуждением от кварцевых резонаторов на основной частоте последовательного резонанса. На выходе схем формируется прямоугольный импульс. Каждый выход кварцевого генератора управляется логическим уровнем:

Лог. 0 – выход не формирует импульсы;

Лог. 1 – выход формирует импульсы.

С помощью управляющих сигналов выходы кварцевых генераторов подключаются к общему выходу устройства.

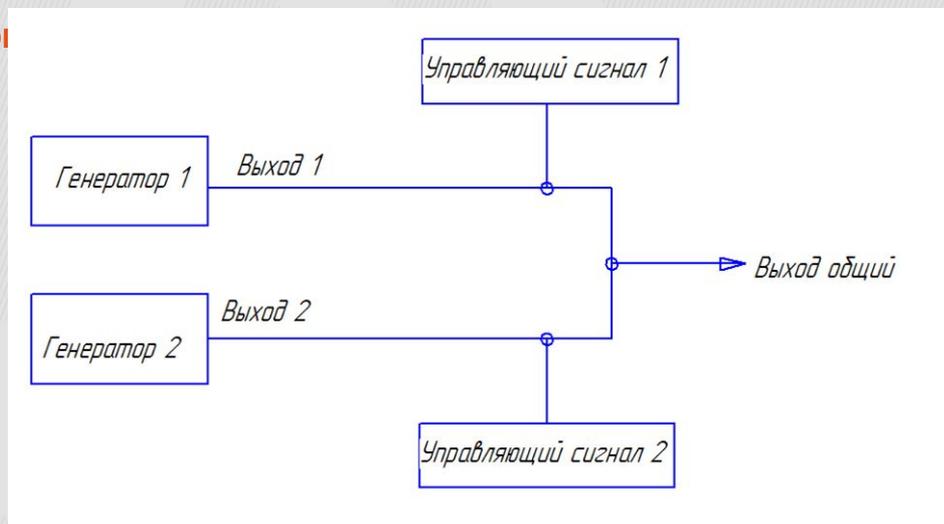
Входные данные:

Напряжение питания постоянного тока 3,3 В;

Частота возбуждения основной гармоники должна лежать в диапазоне от 10 до 100 кГц.

Напряжение логического уровня Лог. 1 3,3В;

Напряжение логическо

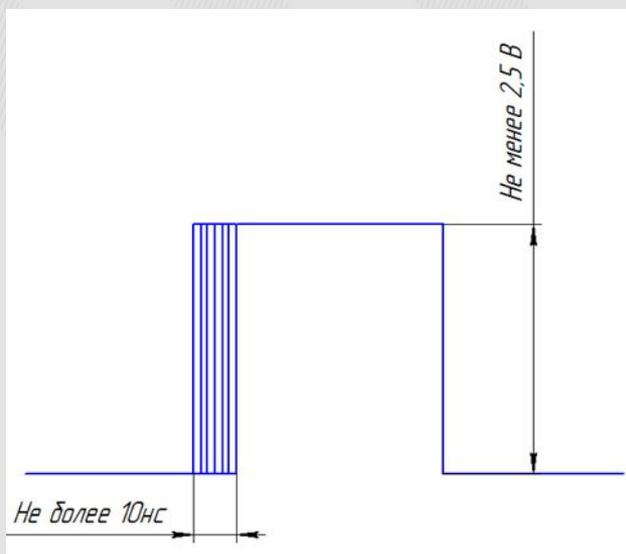


Возможно предоставление кварцевых резонаторов ООО «Радар ммс».

3. Требования к конечному результату:

Выбранные и разработанные схемы должны удовлетворять следующим требованиям:

- амплитуда выходного сигнала: не менее 2,5 В (Рис.2);
- фазовое «дрожание» выходного сигнала (джиттер) менее 10нс (Рис.2).





Колонизация марса

Цель проекта: развитие алгоритмического и абстрактного мышления.

Исходные данные, условия, ограничения:

Ученым удалось отправить на планету Марс мини-фабрику, которая может за одни сутки произвести либо мини-фабрику, либо дрона для сбора воды (мини-фабрика или дрон для сбора воды могут начать работу только на следующие сутки). Дрон собирает одну единицу воды за одни сутки.

Напишите программу, которая определяет, за какое минимальное количество суток удастся собрать не менее N единиц воды.

Требования к конечному результату:

формат входных данных: задается числом N ($1 \leq N \leq 10^9$) – необходимое количество воды;

формат выходных данных: одно целое число M – минимальное необходимое количество суток.

Замечание:

одна из правильных последовательностей действий выглядит так:

в первые сутки мини-фабрика производит дрона для сбора воды;

за вторые сутки мини-фабрика производит еще одного дрона, а первый собирает одну единицу воды;

за третьи сутки мини-фабрика может произвести еще одного дрона или мини-фабрику, при этом первый дрон собирает еще одну единицу воды (итого он собрал 2 единицы воды), а второй дрон собирает единицу воды.

Таким образом, было собрано 3 единицы воды за трое суток.

другая последовательность действий состоит в том, чтобы за первые сутки построить еще одну мини-фабрику, а за вторые сутки произвести двух дронов, которые на третьи сутки соберут 2 единицы воды.



Морской бой

1. Цель проекта: развитие алгоритмического и абстрактного мышления.

2. Исходные данные, условия, ограничения:

Перед началом игры «Морской бой» игрок заполняет игровое поле размером $N \times N$ клеток кораблями $m \times n$. Размер корабля – количество клеток, которое занимает корабль ($m \times n$). Корабли могут иметь только прямоугольную форму (изгибы, например, в виде буквы Г, не допускаются). Корабли не имеют общих и соприкасающихся (сторонами или вершинами) клеток.

Написать программу, которая получает на вход заполненное игроком игровое поле. Результат программы должно быть целое число: средний размер корабля (дробная часть отбрасывается).

3. Требования к конечному результату:

- время выполнения: 1 час;

- формат входных данных: первая строка – размер поля, далее заполненное кораблями игровое поле (где, 1 – занятая кораблем клетка, 0 – свободная).

- формат выходных данных: средний целый размер корабля.

Пример:

Пример:

Входные данные	Выходные данные
7	4
0000000	
0011100	
0011100	
0000000	
0101100	
0101100	
0000000	



Клиент-серверное приложение обмена текстовыми сообщениями

1. Цель проекта: получение навыков разработки клиент-серверных приложений.

2. Исходные данные, условия, ограничения:

разработать клиент-серверное приложение обмена текстовыми сообщениями со следующими требованиями:

- клиентское и серверное приложения должны являться отдельными приложениями;
- клиентское приложение должно быть реализовано в виде GUI (графического пользовательского интерфейса), в котором:
 - необходимо создать графическую форму, в которой будет осуществляться обмен текстовыми сообщениями с другим пользователем (далее с контактом);
 - необходимо создать отдельную форму или обособленный графический элемент на общей форме с полем авторизации, в котором пользователь будет задавать свой никнейм;
 - необходимо создать отдельную форму или обособленный графический элемент для добавления других контактов в адресную книгу и для перехода в активный диалог с контактом;
 - клиентское приложение должно хранить данные о контактах, историю переписок с контактами во встроенном хранилище;



- серверное приложение должно быть реализовано в виде консольного приложения;
- передача сообщений должна осуществляться через стек TCP/IP;
- серверное приложение должно хранить все контакты и все истории переписок между контактами;
- передача сообщений должна быть защищена шифрованием (опционально, пункт повышенной сложности);

3. Требования к конечному результату:

- при реализации шифрования сообщений необходимо использовать существующие (общепринятые) реализации алгоритмов.

5. Возрастная категория: студенты средних и высших учебных заведений.



Таск-менеджер

1. Цель проекта: получение навыков разработки desktop-приложений.

2. Исходные данные, условия, ограничения:

таск-менеджер – программа, которая помогает управлять проектами, планировать задачи. Необходимо:

- разработать приложение с графическим пользовательским интерфейсом (GUI) для планирования задач с напоминанием по дате/времени;
- реализовать графическую форму для вывода на экран всех задач, предусмотреть возможность сортировки по приоритетам;
- реализовать возможность добавления, изменения, удаления задач;
- необходимо реализовать механизм вывода уведомлений на экран;
- приложение должно хранить информацию о задачах во внутреннем хранилище;

3. Требования к конечному результату:

- время выполнения: 3 дня;

4. Возрастная категория: студенты средних и высших учебных заведений.



Кейс 1 «Аварийная посадка БАС»

С популяризацией и широким применением беспилотных авиационных систем (БАС) возрастают требования к безопасности для обеспечения целостности как используемых устройств, так и человеческих жизней.

Автономная посадка является фундаментальным аспектом эксплуатации БАС. Для осуществления безопасной посадки, появляется необходимость в автоматическом распознавании и построении маршрута к подходящей ближайшей точке приземления. Таким образом, существует потребность в разработке систем технического зрения, позволяющих определить местоположение БАС для автоматической системы реагирования на чрезвычайные ситуации, которая способна постоянно контролировать окружающую среду и при необходимости определять безопасные места посадки (или даже падения). Применение подобных систем реагирования должно значительно снизить риски при выполнении полетных заданий, что может стать средством обеспечения безопасности для многих приложений, включая решение задач сельского хозяйства, мониторинг, транспортировку, поиск и спасение.

Ожидаемый результат:

Программное решение на основе ИНС, предназначенное для систем технического зрения, позволяющих БАС ориентироваться на местности и определять безопасные точки приземления.

Исходные данные:

Для осуществления ориентации и навигации БАС оснащён дополнительными камерами, захватывающими область местности, находящуюся под ним. Данные камеры позволяют получить кадры, представляющие собой RGB изображения и изображения в ближнем инфракрасном диапазоне (БИК). Для обучения алгоритма распознавания сцен предлагается использовать набор данных GID, состоящий из аннотированных изображений (RGB + БИК), в которых размечены 15 классов местности, такие как: промышленная зона, городская жилая зона, сельская жилая зона, дороги, лес, рисовое поле, орошаемая земля, сухие пахотные земли, садовый участок, кустарниковая степь, естественное пастбище, искусственная трава, река, озеро, пруд.

Постановка задачи:

Необходимо разработать алгоритм распознавания сцен местности, а именно:

- выбрать или разработать подходящую архитектуру искусственной нейронной сети (ИНС);
- обучить модель ИНС на основе выбранной архитектуры;
- сформировать и реализовать предложения по оптимизации обученной ИНС для использования на вычислительных системах малой мощности.



Кейс 2 «БАС в сельском хозяйстве»

БАС используются во многих областях сельского хозяйства включая обработку полей, распыление пестицидов и удобрений, точное земледелие и т.д. Применение БАС позволяет повысить эффективность проведения технологических процессов в агропромышленности. Чтобы обеспечить безопасную и надёжную автономную работу БАС, выполняющего задачу обработки полей, необходимо использовать интеллектуальную систему автономного принятия решений, которая предназначена для построения актуального полетного задания, характеризующего эффективный облет поля.

Ожидаемый результат:

Программное решение на основе ИНС, предназначенное для интеллектуальных систем автономного принятия решений, позволяющих БАС автоматически формировать полетное задание для облета поля при решении задачи обработки сельскохозяйственных полей.

Формат применения кейс-задания:

Задание для организации хакатонов, конкурсов и конференций для студентов средних и высших учебных заведений.

Исходные данные:

БАС оснащены камерами, позволяющими получать изображения в формате RGB. Камеры захватывают кадры с участками земли, над которыми пролетает БАС. Для обучения ИНС сегментированию сельскохозяйственных полей предлагается использовать и аннотировать изображения участков местности Томской области, полученные с помощью картографического сервиса Google Карты. Пример аннотирования изображения представлен на рисунке ниже.



Постановка задачи:

Необходимо разработать элемент интеллектуальной системы автономного принятия решений, которая предназначена для построения актуального пути передвижения БАС (облета поля), выполняющего задачу обработки сельскохозяйственных полей. Таким образом, необходимо:

- создать аннотированный набор данных для обучения;
- определить и обучить модель ИНС, предназначенную для сегментации сельскохозяйственных полей.

Примечание: полученная при сегментации поля маска и является базой для формирования полетного задания;

- проработать варианты интеграции полученной маски на выходе ИНС с любым полетным контроллером открытой архитектуры.

Название работы: Программа для шифрования и расшифровка текста.

Цель: Разработать программу для шифрования и расшифровки текста.

Языки программирования: Python, C++, C#, Java, Delphi и др.

Требования к программе:

Программа должна поддерживать один или несколько методов шифрования.

Метод шифрования выбирается на усмотрение участников. При выборе существующего метода шифрования требуется обоснование его выбора, при разработке собственного – обоснование выбора и принцип его работы.

Программа должна иметь функционал как для шифрования, так и для расшифровки сообщений, которые были зашифрованы с помощью этой же программы и этого же метода шифрования.

Пользователь должен иметь возможность вводить собственные текстовые сообщения, а также указывать параметры шифрования (например, ключ или ключевое слово).

Программа должна иметь интуитивно понятный пользовательский интерфейс.

Дополнительные функции, такие как дополнительные методы шифрования, импорт/экспорт зашифрованных сообщений, «защита от дурака» и т.д. будут оценены как преимущество.

Членами комиссии будет произведена оценка проекта по следующим критериям:

Функциональность и эффективность программы;

Оригинальность и инновационность подхода.

Название работы: Разработка и внедрение методики оценивания знаний офисных работников.

Цель: Разработка и внедрение методики оценки знаний пользователей по направлениям:

Базовые знания операционных систем;

Знания офисного ПО;

Информационная безопасность.

Условия задачи:

На предприятии «Компас ++» в качестве операционных систем используются:

Windows 10;

Windows 8.1;

Astra Linux SE.

Из офисного ПО затронуты:

MS Office;

Libre Office.

Требуется разработать методику оценивания знаний пользователей в области владения компьютером, система оценивания которой должна быть объективной и определять уровень владения ПК пользователем.

Методика должна выявлять слабые места в знаниях пользователей по заданному направлению и выдавать рекомендации по устранению пробелов в знаниях пользователя.



Требования к методике:

Объективность. Методика должна быть объективной и надёжной, результаты оценки должны быть справедливыми и не зависеть от личных предпочтений или субъективных факторов.

Адаптивность. Методика должна быть адаптивной к целевой аудитории, учитывая их уровень знаний и навыков работы с ПК.

Репрезентативность. Методика должна оценивать не только базовые знания работы с ПК, но и более сложные задачи, которые могут возникнуть в реальных условиях работы.

Надежность. Методика должна быть достаточно точной, чтобы её результаты можно было использовать для принятия важных решений в организации.

Краткость. Методика должна быть краткой и легко понятной, чтобы она могла быть использована без особой подготовки и не занимала много времени.

Удобство. Методика должна быть удобной в использовании как для экспертов, проводящих оценку, так и для пользователей, проходящих тестирование.

Доступность. Методика должна быть доступна и распространена, чтобы ее мог использовать широкий круг подразделений организации.

Практичность. Методика должна быть практичной и иметь практическое применение в реальных условиях работы, чтобы она могла быть использована для повышения уровня знаний и навыков работы.

Критерии оценки знаний пользователя должны включать оценку:

- Объема знаний
- Качества знаний
- Глубину знаний
- Опыт

Ожидаемый результат: методика, включающая в себя как минимум тест/тестовое задание/интервью с пользователем.



Название работы: Применение социальной инженерии для оценки защищенности предприятия.

Цель: Использовать методы социальной инженерии для разработки методики оценки защищенности предприятия от атак социальной инженерии в будущем.

Исходные данные: <https://www.kaspersky.ru/resource-center/threats/how-to-avoid-social-engineering-attacks>, https://ru.wikipedia.org/wiki/Социальная_инженерия, другие статьи в Интернете

Условия и ограничения:

У Вас имеется штат сотрудников в 100 человек разных возрастов, специализации и опыта работы с ПК. Вам нужно реализовать систему проверки и оценивания уровня защищенности этих сотрудников в случае применения в отношении них атак социальной инженерии. Нужно, чтобы эта система была понятна и удобна в использовании, подразумевала периодическую проверку в заданных вами промежутках, а ее результаты можно было легко интерпретировать. Данная система должна также включать себя полную инструкцию по использованию для исполнителей.

Конечный результат: методика, которая построена на использовании почтовой системы/электронных носителей информации/телефонной связи, может включать в себя презентацию/тест/программу.

Название работы: Программа для шифрования и дешифрования текста.

Цель: Разработать программу для шифрования и расшифровки текста с графическим интерфейсом пользователя (GUI).

Задание: Разработайте программу для шифрования и расшифровки текста с графическим интерфейсом пользователя (GUI).

Языки программирования: Python, C++, C#, Java, Delphi и др.

Требования к программе:

1. Программа должна поддерживать несколько методов шифрования. Каждый метод должен быть реализован в отдельном модуле/функции.
2. Методы шифрования выбираются на усмотрение участников. При выборе существующего метода шифрования требуется обоснование его выбора, при разработке собственного – обоснование выбора и принцип его работы.
3. Программа должна иметь функционал как для шифрования, так и для расшифровки сообщений, которые были зашифрованы с помощью этой же программы и этого же метода шифрования.
4. Пользователь должен иметь возможность вводить собственные текстовые сообщения, а также выбирать метод шифрования и указывать параметры шифрования (например, ключ или ключевую фразу).
5. Программа должна поддерживать функцию импорта/экспорта зашифрованных сообщений в форматах, выбранных участниками.



6. Программа должна иметь интуитивно понятный пользовательский интерфейс, который будет доступен даже непрофессиональным пользователям, а также «защиту от дурака».

Членами комиссии будет произведена оценка проекта по следующим критериям:

- Функциональность и эффективность программы;
- Качество пользовательского интерфейса и удобство его использования;
- Оригинальность и инновационность подхода.



Название работы: Программа для безопасной передачи сообщений.

Цель: Разработать программу для безопасной передачи сообщений, которое позволит пользователям отправлять и получать сообщения с использованием алгоритмов шифрования.

Задание: Разработайте программу для безопасной передачи текстовых сообщений по сети, которое позволит пользователям отправлять и получать сообщения с использованием алгоритмов шифрования.

Время выполнения: 2 дня.

Языки программирования: Python, C++, C#, Java, Delphi и др.

Требования к программе:

1. Программа должна иметь клиент-серверную архитектуру.
2. В программе должна быть реализована возможность создания чатов с другими пользователями и передачи сообщений внутри них.
3. Программа должна поддерживать отправку файлов.
4. Программа должна иметь функции шифрования и расшифровки сообщений и файлов, проверки целостности сообщений. Метод шифрования выбирается на усмотрение участников (с последующим обоснованием выбора).

- 
1. Программа должна иметь интуитивно понятный пользовательский интерфейс, который будет понятен даже непрофессиональным пользователям, а также «защиту от дурака».
 2. Дополнительные функции, такие как поддержка групповых чатов, поддержка различных операционных систем и устройств и т.д. будут оценены как преимущество.

Участники должны будут разделиться на команды (от 2 до 5 человек) и создать программу в соответствии с требованиями. Каждая команда должна представить свой проект в конце мероприятия. Членами комиссии будет произведена оценка проекта по следующим критериям:

- Функциональность, эффективность и безопасность программы;
- Качество пользовательского интерфейса и удобство его использования;
- Оригинальность и инновационность подхода.

Возрастная категория: студенты ВУЗов



Название работы: Разработка приложения для централизованного хранения паролей пользователей.

Цель: Разработать приложение для централизованного хранения паролей пользователей с доступом к нему с различных устройств с помощью единого пароля.

Задача: Требуется разработать приложение для централизованного хранения паролей пользователей с возможностью доступа к нему с различных устройств. Также должна решаться задача создания надежного пароля (генератор паролей). Решение должно обеспечивать возможность одновременного многопользовательского подключения.

Требования:

1. **Безопасность.** Система хранения паролей должна быть безопасной, чтобы предотвратить возможность несанкционированного доступа к паролям.
2. **Шифрование.** Все хранимые пароли должны быть зашифрованы, чтобы предотвратить возможность их раскрытия.
3. **Контроль доступа.** Система должна иметь механизмы контроля доступа, чтобы обеспечить разграничение прав доступа к хранимым паролям, а также логирование всех действий пользователей.

- 
4. **Парольная политика.** Система должна предоставлять возможность создания и управления парольной политикой.
 5. **Автоматическая генерация паролей.** Система должна предоставлять возможность автоматической генерации надежных паролей, которые сложно подобрать.
 6. **Управление жизненным циклом паролей.** Система должна предоставлять механизмы управления жизненным циклом паролей, такие как установка срока действия паролей, напоминания о смене паролей, автоматическая блокировка учетной записи после определенного числа неудачных попыток.
 7. **Возможность делиться паролем.** Система должна иметь механизмы создания пароля, к которому будет иметь доступ заданный круг лиц.
 8. **Резервное копирование.** Система должна иметь возможность создания резервной копии хранимых паролей, чтобы обеспечить восстановление доступа к учетным записям в случае сбоя в системе.
 9. **Поддержка множества пользователей.** Система должна быть способной поддерживать множество пользователей и учетных записей, а также обеспечивать доступ к паролям только тем, кто имеет соответствующее права.

Результатом выполнения задания может являться: клиент-серверная программа.

Возрастная категория: студенты ВУЗов



Название проекта: моделирование траектории движения судна для обследования морского дна

Цель проекта: разработать приложение, моделирующее оптимальную траекторию движения судна для обследования морского дна в заданной области с учетом запретных зон и зон усиленного сканирования. В приложении должен быть реализован функционал рисования области исследования в виде полигона, с добавлением зон.

2) Исходные данные, условия, ограничения:

Необходимо выполнение следующих условий:

При моделировании траектории вся зона обследования должна быть покрыта зоной сканирования. Приложение должно строить траектории для следующих типов зон сканирования:

- 3 метра от левого и правого борта судна;
- 6 метров от правого борта;
- 5 метров от левого борта.

Зона сканирования не должна пересекаться с запретными зонами. Зона усиленного сканирования должна быть обследована более детально (дважды). Траектория должна иметь наименьшую протяженность и время выполнения. Приложение должно строить траектории для следующих характеристик судна:

- максимальная скорость судна – 16 км/ч, радиус разворота при максимальной скорости – 4 м;
- максимальная скорость судна – 4 км/ч, радиус разворота при максимальной скорости – 6 м;

Радиус разворота линейно пропорционален скорости судна. Полигон может состоять из прямых линий под любым углом;

4) Требования к конечному результату, программному обеспечению, используемым материалам, длительности выполнения задания:

Программное обеспечение должно запускаться без установки дополнительных библиотек и дополнительного программного обеспечения.

Используемое программное обеспечение: программное обеспечение по выбору студента и согласованию с куратором.

Используемые материалы: научные сборники, статьи, доклады и результаты анализа научных материалов.



1) Название проекта: разработка дизайн-проекта интерьера катера.

Цель проекта: проведение исследований и разработка эскизов нового дизайн-проекта интерьера катера.

2) Исходные данные, условия, ограничения: примеры чертежей внутренней части катера, готовые решения интерьеров судов гражданского назначения.

4) Требования к конечному результату, программному обеспечению, используемым материалам, длительности выполнения задания:

Интерьер катера включает в свой состав следующие элементы:

- Кают-компания:

- ниши для книг, журналов и прочего;
- место для подогрева пищи (минихолодильник, смеситель и раковина, электроплитка и микроволновая печь);
- диван;
- складываемый столик;
- блок из 3 розеток по 220В;

Галюун:

- раковина с тумбой для хранения ваннх принадлежностей;
- ниши для хранения чистящих средств;
- унитаз с электрическим насосом;

Душевая:

- раковина с тумбой для хранения ваннх принадлежностей;
- душевая с 3х секционной дверью, отделяющей душевую кабину;

Каюта экипажа №1, 2:

- койка с выдвжными ящиками для хранения вещей;
- второе спальное место, складываемое;
- двухсекционный вентилируемый шкаф для хранения вещей;
- ниша для книг, журналов и прочего;

Мастер-каюта:

- двухсекционный шкаф для хранения вещей;
- ниша для книг, журналов и прочего;
- стол на телескопической опоре;
- диван, трансформирующийся в угловую кровать;
- иллюминатор.

Примечания:

- каюта должна иметь несколько глухих иллюминаторов;
- из каюты должен быть предусмотрен аварийный выход в рубку размером не менее 600х600 мм.

Высота в каюте в проходах не менее 1900 мм;

- все койки, кроме койки-дивана в каюте, должны иметь ширину в средней части не менее 580 мм;
- на всех койках должна быть обеспечена возможность выпрямиться в сидячем положении;
- из тамбура должен быть обеспечен вход в помещение галюона, в каюту и сушилку.

При выполнении проекта должны быть решены следующие задачи:
эргономика: удобство расположения панелей управления, количество, тип и расположение кресел, панорамное остекление, система вентиляции и т.д.)

Разработать эскиз интерьера катера:

Цель: для размещения элементов в помещениях;

Формат: рисунок на бумажном носителе или в электронном виде;

Разработать дизайн-проект катера:

Формат: электронный вид 3D моделей помещений и скриншоты видов;

Цвет: на усмотрение исполнителя.

Предоставить отчетную документацию:

Отчет по изготовлению дизайн-проекта:

на бумажном носителе в двух экземплярах в ледериновом или Unibind-переплете;

на электронном носителе (CD-R или DVD-R) в одном экземпляре в формате doc и pdf.

Демонстрационные материалы для презентации результатов работы:

формат презентации PowerPoint – pptx;

тип электронного носителя презентации – DVD-R;

количество экземпляров – 1 шт.

Требования к конечному результату: Реализованный эскиз

Используемое программное обеспечение: 3D max, на выбор школьника.

Используемые материалы: книги по судостроению, по выбору школьника.



Название проекта: разработка системы оптической фокусировки светового (лазерного) потока для формирования луча с фиксированной областью засветки (например, 5 градусов)

Цель проекта: Разработка оптической системы формирования светового (лазерного) потока с заданной итоговой областью засветки

Формат применения кейс-задания: для организации кейс-фестиваля «Будущее Петербурга – в творчестве юных» (школьники, учащиеся лицеев).

Исходные данные, условия, ограничения:

Учебники по геометрии (раздел Углы, прямые), физика (раздел Оптика (падение, отражение лучей и среды преломления лучей)), требования по освещенности поверхности.

Требования к конечному результату, программному обеспечению, используемым материалам, длительности выполнения задания:

Требования к конечному результату: фиксированная оптическая система, позволяющая формировать световой (лазерный) поток с заданной областью засветки при поступлении луча света в систему

Используемое программное обеспечение: TracePro.

Используемые материалы: оптические элементы (линзы, зеркала), устройства для калибровки установленных оптических элементов.