1. Створити таблицю в середовищі Builder C++

Марка автомобіля | Країна виробник | Вага | Об'єм двигуна | Потужність | Ціна

1. Відомим алгоритмом відсортувати автомобілі по ціні
2. Вивести список автомобілів у заданому ціновому проміжку
3. Визначити однакові автомобілі за співвідношенням Потужність/Вага
4. Визначити та відобразити найгірший та найкращий автомобіль за співвідношенням Потужність/Вага до ціни
5. Порахувати середній об’єм двигуна по кожній країні виробника
6. Визначити найефективніший автомобіль за співвідношенням Об'єм двигуна до співвідношення потужність/вага
7. Написати програму у середовищі Builder C++ для обліку персонального бюджету, яка передбачає декілька джерел доходів (заробітна плата, дохід по депозиту, і т.д ), а також декілька джерел видатків (оплата помешкання, витрати на харчування, розваги). Джерела видатків можуть оподатковуватися, відсоток оподаткування визначається при створенні джерела видатку. При переміщенні суми грошей з доходів у видатки, сума податку знімається додатково з джерела доходу автоматично. Список усіх транзакцій повинен зберігатися та зчитуватися з файлу budget.dat
8. Створити таблицю у середовищі Builder C++:

Найменування плитки | Ширина плитки | Висота плитки | Ціна за 1м.кв.

1. Порахувати та відсортувати плитки відомим алгоритмом за ціною однієї плитки
2. Вивести кількість плиток у одному 1м.кв. для обраного найменування плитки з заокругленням завжди у більшу сторону
3. Розрахувати кількість плиток для стіни, параметри якої (ширина та висота) задаються користувачем
4. Розрахувати вартість плиток для визначеної стіни, при умові:
   1. Плитки можна купувати по-штучно
   2. Плитки продаються лише по м.кв
5. Написати програму у середовищі Builder C++ для побудови та відображення Генеалогічного дерева. Кожен елемент дерева повинен містити наступні властивості, які задаються користувачем при створені елемента: Ім'я, Прізвище, Стать та Дата народження. Данні дерева повинні зберігатися та вичитуватися з фалу tree.dat.
6. Створити програму у середовищі Builder C++ для обліку працівників у компанії. Запис про працівника повинен мітити наступні поля:

Прізвище | Ім’я | День народження | Дата працевлаштування | Відділ | Проект | Заробітна плата

1. Відомим алгоритмом відсортувати працівників по даті працевлаштування у компанію
2. Відобразити всі проекти у визначеному відділі
3. Визначити та відобразити відділи, в яких працівники мають день народження у визначеній даті
4. Показати проект, в якому працює наймолодший та найстарший працівник
5. Відсортувати відомим алгоритмом проекти по середній заробітній платні працівника
6. Створити програму для товарів на складі. Кожен товар описується наступними властивостями:

Назва | Тип | Кількість одиниць | Дата виготовлення | Термін зберігання | Ці на за одиницю

1. Відобразити тип товару з найменшим середнім терміном зберігання
2. Показати товари, термін придатності яких виходить у заданому користувачем місяці
3. Розрахувати та відобразити сумарну вартість товарів по типам, відсортувати типи товарів по сумарній вартості, використовуючи відомий алгоритм
4. Вивести список товарів у визначеному діапазоні ціни за одиницю. Діапазон визначається користувачем

Аплікація повинна зберігати та читати усі внесенні товари з файлів, назва яких співпадає з типом товару.

1. Написати програму для аналізу текстів. Тексти можуть вводитися користувачем через клавіатуру або ж зчитуватися з файлу. Програма повинна мати наступну функціональність:
2. Виводити відсортовану по кількості таблицю з літерами латинського алфавіту у форматі :

Назва літери | Кількість у тексті

Літери, які не зустрічаються у тексті вносити у таблицю не потрібно.

1. Реалізувати функціонал описаний в першому пункті, але для всіх символів, які зустрічаються в тексті
2. Реалізувати можливість калькуляції слів у тексті. Словом вважати будь-яку послідовність символів, відділену знаком пробілу
3. Відобразити десять найбільш вживаних у тексті слів
4. Реалізувати функціонал заміни одного слова на інше у тексті, з можливістю збереження модифікованого тексту у новий файл
5. Написати аплікацію для зберігання каталогу мобільних пристроїв (телефонів та смартфонів)
6. Створити клас “Smartphone” для збереження інформації про смартфон:
   1. Рік виробництва
   2. Виробник
   3. Модель
   4. Розмір пам’яті
   5. Діагональ екрану
   6. Колір
7. Створити клас “Phone” для збереження інформації про телефон:
   1. Рік виробництва
   2. Виробник
   3. Модель
   4. Колір
   5. Кількість сім-карт
8. Для цих класів створити конструктор за замовчуванням, конструктор з параметрами, конструктор копій; перевизначити операції >>, << для зчитування та запису у файл.
9. Реалізувати функцію для пошуку мобільних пристроїв по назві виробника та роком виробництва.
10. Визначити та вивести мобільні пристрої, з однаковим кольором та роком виробництва
11. Інформація про мобільні пристрої повинна зберігатися та зчитуватися у файли smartphones.dat та mobilephones.dat відповідно
12. Написати програму для обчислення площі та периметру плоских математичних фігур (прямокутник, трапеція, трикутник та правильний шестикутник). Програма повинна автоматично відображати поля, які потрібно заповнити для обчислення площі та периметру в залежності від типу фігури, яку обрав користувач. Програма повинна вести історію по кожній з фігур та записувати/зчитувати історію з файлу.
13. Написати програму для створення RSS 1.0 файлу новин. Новина описується наступними параметрами: Заголовок, Посилання, Опис. Програма повинна зчитувати усі новини з заданого RSS файлу, додавати або оновлювати новини, записувати зміни у RSS файл. Додатково програма має реалізовувати функціонал статистики по RSS файлу: кількість новин, новина з найбільшим описом, Новина з найкоротшим заголовком. При збереженні файлу дублікати новин повинні видалятися автоматично.
14. У середовищі Builder C++ створити програму для обліку витрат по кожній з квартир у багатоквартирному будинку. Кількість квартир визначається статично. Розрахунковий період – 1 місяць. Інформація про квартиру:

Номер квартири | Прізвище власника | Спожито води (кубометрів) | Спожито світла (кВт⋅год)

Функціональність програми:

1. Можливість внесення інформації по кожній з квартир для визначеного користувачем місяця. Відображення місяців, для яких інформація не заповнена для кожної з квартир
2. Збереження та зчитування річної інформації з файлу, назва файлу відповідає року.
3. Можливість встановлення ціни в межах року за один кубометр води та за одну кВт⋅год. Ціна записується у файл року. Розрахунок та відображення вартості спожитих послуг по місячно, агреговану для всіх квартир
4. Можливість завантаження у програму більше одного року
5. Вивести власника, який за рік спожив найменше послуг в грошовому еквіваленті. Виведення номеру квартири, яка спожила найбільше послуг у грошовому еквіваленті в межах усіх років, які поточно завантаженні у програму
6. Створити таблицю в середовищі Builder C++

Прізвище | Ім‘я | Кафедра | Група | Середній бал | Семестр | Рік

1. Відобразити відсортований відомим алгоритмом список студенів за середнім балом, відповідно до вибраного року та семестру
2. Відобразити найбільш та найменш успішніші роки за середнім балом
3. Побудувати графік успішності засобами Builder C++ для вибраного студента
4. Відобразити студентів у вибраному діапазоні середнього балу, семестрів та років
5. Відобразити список з п’ятьох найкращих студенів
6. Визначати групу, яка була найуспішніша у вибраному році
7. Відобразити студентів, успішність яких лише протягом їх навчання
8. **Аквапарк.**

Створити таблицю в середовищі Builder C++

№ | Прізвище | Ім‘я | Телефон | Тип абонаменту | Строк дії абонаменту

За заданим номером визначити cтрок дії абонаменту (врахувати наявність в таблиці однакових прізвищ)

1. За прізвищем визначити телефон, тип абонаменту та його строк дії
2. За телефоном визначити прізвище та ім’я
3. Знайти імена та прізвища людей, у яких однаковий тип абонаменту.
4. Визначити тип абонаменту який має набільшу кількість записів
5. Вивести всі прізвища які мають однакове значення «Строк дії абонаменту».
6. **Батальйон.**

Створити таблицю в середовищі Builder C++

№ | Прізвище | Ім‘я | вік | група крові | звання | боєкомплект |

1. Порахувати кількість одиниць озброєння виданого бійцям, зрупувавши бійців по званню.
2. Визначити середній вік бійців для кожного звання і вцілому по батальйону.
3. Вивести бійців які мають більше однієї одиниці зброї, бійців які мають АК-47 і гранатомет, які мають пістолет і будь-який автомат.
4. Вивести бійців які мають першу групу крові резус мінус і четверту групу резус плюс віком до 25 років.
5. Вивести рядових віком до 20 років без боєкомплекту.
6. **Дрон.**

Створити таблицю в середовищі Builder C++

№ | Модель | Оператор | Дистанція польоту | Вистота | Швидкість | Статус

1. Відсортувати за найкращим показником «Швидкість», «Дистанція польоту»
2. Згрупувати по моделях які мають результат «Статус» – «успішне повернення» та найкраще значення – «Швидкість». Відсортувати за Оператором.
3. Визначити моделі, які мають приблизно однаковий результат за показником «Висота» з відхиленням 7,359%
4. Визначити середній результат «Дистанція» та знайти операторів, у яких результат «Статус» - «потрачено», та визначити відхилення їх результату «Статус» від середнього значення.
5. Вивести операторів, у яких загальний результат буде найкращим (серед тих які управляють однаковою моделлю дрона).
6. **Персональний Бюджет**

Створити таблицю в середовищі Builder C++

№ | Дата | Час | Тип витрат | Підтип витрат | Сума витрат | Валюта витрат | Курс

1. Відомим алгоритмом відсортувати записи таблиці за значенням дати; типом і підтипом витрат; валютою витрат; сумою витарат в гривнях.
2. Вивести 3 дати, що мають найбільші витрати в доларах США.
3. Поділити добу на проміжки по 6 годин і визначити в які проміжки найбільше значних витрат, дрібних витрат, витрат всього.
4. Визначити скільки разів на місяць здійснюється в середньому витрата «харчування» розміром менше 20 грн.
5. Вивести тип і підтип 5 найдрібніших витрат
6. Вивести витрати у дні коли змінювався курс валют.
7. **Вибори**

Створити таблицю в середовищі Builder C++

№ | Виборчий округ | Витрати на навчання | Кількість спостерігачів | Кількість членів ДВК

1. Відомим алгоритмом відсортувати записи за кількістю членів ДВК в округах.
2. За заданим округом визначити середні витарати на навчання на одну людину (спостерігачі і члени ДВК)
3. Вивести округи в яких спостерігачів менше 50% від кількості членів ДВК
4. Визначити округи в яких витрати на члена комісії і ДВК менші за середні.
5. Вивести 5-ть округів в яких кількість спостерігачів більша середньої, проте витрати на навчання одного спостерігача менші середніх.
6. **Марафонський біг**

Створити таблицю в середовищі Builder C++

№ | Прізвище | Ім‘я | Рік народж. | Стать | Пульс | Дистанція | Час завершення (для тих хто вже завершив забіг).

1. Відомим алгоритмом відсортувати записи за показником дистанція для кожної статі,
2. Серед жінок визначити тих, в кого пульс вище середнього серед жінок.
3. Серед чоловіків визначити наймолодшого з дистанцією більше середньої серед чоловіків та пульсом в межах 20% від середнього серед чоловіків.
4. Визначити всіх людей старших за 60 років, в кого пульс вище 90.
5. Всім бігунам, вік яких менший, ніж 28, з показником пульсу нижче 120 вивести повідомлення «---Прізвище--- так триматись!»
6. Всім бігунам в кого дистанція > 40 км але менше дистанції марафону вивести повідомлення «Скоро фініш»
7. Бігунам в кого дистанція найбільша серед своєї групи (жінки/чоловіки у вікових групах до 28, до 45, до 60 і більше 60 років) вивести повідомлення «Ви Лідер!»
8. Бігунам які досягнули фінішу вивести їх результат: « Вітаємо \_\_Прізвище\_\_ з завершенням забігу, ваш результат \_\_час завершення забігу\_\_, місце \_\_місце\_\_\_».
9. **Довідник з геомертії**

Написати програму з можливістю вибору типу і підтипу геометричної фігури. Після вибору моделі і її розміру програма повинна намалювати її та вивести інформацію про назву фігури, площа, периметр, радіус, довжина сторін, розмір кутів, радіус описного та в вписаного кола, довжину середньої лінії та ін. Необхідну інформацію про типи і підтипи фігур взяти з <http://en.wikipedia.org/wiki/List_of_mathematical_shapes>

1. **Розумний дім**

Створити таблицю в середовищі Builder C++

Рік | Місяць | Число | t | Тиск | Вологість | Витрати електроенергії

1. Визначити середню вологість за місяць
2. Визначити дні з найвищою t та найменшим тиском
3. Визначити дні, протягом яких витрати електроенергії більше 5 kw
4. За заданим періодом, визначити середню температуру протягом цього періоду та дні з найвищою вологістю.
5. Знайти період, в якому тиск змінювався в межах ±1,43%, а t – 5,673%.
6. Побудувати засобами Builder графіки t, тиску та вологості
7. Спрогнозувати витрати електроенергії на наступний місяць врахувавши зростання витрат на попередні три місяці по відношенню до аналогічних трьох місяців попереднього року.
8. **Таксі**

Створити таблицю в середовищі Builder C++

№ авто | Прізвище водія | Вартість кілометра | Вартість авто | Модель авто | Час прибуття | Відстань

1. Відомими алгоритмом відсортувати записи за Вартістю маршруту.
2. Визначити № авто з найменшим часом прибуття.
3. Вивести Прізвище водія, клас і модель авто, вартість перевезення для обраного № автомобіля.
4. Визначити, найдорожчі авто з найнижчою вартістю проїзду і найменшим часом прибуття.
5. Згрупувати записи, в яких вартість проїзду і час прибуття співпадають .
6. **Оптимізація бізнесу**

Розробити ієрархію класів: базовий клас «Торгівельний об’єкт» (Назва, адреса, дата створення, середня виручка, середній прибуток, назва товарів, кількість кожного типу товару); похідні класи «Магазин звичайний» (директор, кількість працівників), «Автоматичний кіоск» (відповідальний оператор, дата обслуговування). Для кожного класу створити: 1) конструктор за замовчуванням; 2) конструктор з параметрами; 3) конструктор копій; 4) перевизначити операції >>, << для зчитування та запису у файл.

Наповнити класи даними, прочитавши відповідні torhivelnyj\_objekt.dat, magazyn.dat, avtomatychnyj\_kiosk.dat. Реалізувати функції пошуку торгівельного об’єкту за назвою, адресою, датою створення. Сортувати торгові точки по адресах та даті створення. Вивести кількість працівників магазину з найбільшою виручкою. Вивести оператора кіоску з найбільшою кількістю товарів. Вивести всі кіоски які мають більше одного оператора та всіх операторів які мають більше одного кіоску. Вивести всі кіоски і магазини які знаходяться на однойменних вулицях. Вивести операторів які обслуговують кіоски рідше ніж раз в 28 годин. Для демонстрації роботи програми використати засоби Builder C++.