**Задача 1.**

Да се дефинира клас Expr, чрез който да могат да се намират стойностите на изразите

S1 = 1 + x / 1! + (x^2) / 2! + … + (x^n) / n!

S2 = 1 – (x^2) / 2! + (x^4) / 4! + ... + (-1)^n (x^(2n)) / (2n)!

Пояснение:

Класът да съдържа две полета, които да пазят x и n.

Да се реализират конструктор и 2 член-функции.

**Задача 2.**

Да се дефинира клас BankAccount, определящ банковата сметка на клиент, съдържаща : име на клиент (символен низ с дължина до 23 знака), номер на банковата сметка (символен низ с дължина до 15 знака) и наличната сума на клиент (реално число). Класът да притежава методи, чрез които : да създава банкова сметка на клиент, да извежда на екрана банкова сметка, да внася и тегли пари от банкова сметка. Да се дефинира главна функция, която да демонстрира класа.

**Задача 3.**

Лекар трябва да поддържа статистика за броя на посещенията, които са направили пациентите му. За всеки пациент лекарят записва името, датата на раждане и броя посещения, които е направил пациентът пред една календарна година. Да се напише програма, която за лекар създава списък от пациентите му, намира средния брой на посещенията на пациентите, както и пациентите, направили повече от зададен брой посещения.

Пояснение :

Да се реализират :

Клас Date, който има три полета – ден, месец, година ; конструктор с подразбиращи се стойности, гетъри и сетъри и метод за принтиране на датата.

Клас Patient, който има 3 полета – име, дата и брой посещения както и съответните им гетъри заедно с метод за принтиране. Освен това да се реализират default-ен конструктор и такъв с параметри.

Клас Doctor, който има 3 полета – име, масив с пациенти (статичен) и брой на регистрираните пациенти. Също така да има методи : конструктор с параметър (името на доктора) , метод за регистриране на пациент ( добавяне към масива с регистрирани пациенти), метод average(), който връща средния брой на посещенията на пациентите, метод printUpper(int), който принтира само тези пациенти, чийто брой визите е повече от подадения като параметър на функцията.