***Примерни задачи за изпит***

1. Да се напише функция на C++, която приема като параметър два едномерни сортирани целочислени масива с елементи от тип int и връща като резултат трети масив (който съдържа всички елементи на двата масива и само тях), който също е сортиран. Предполага се, че входните масиви са сортирани преди да се извика написаната от вас функция.
2. Да се напише функция на C++, която приема като параметри две отсечки в декартовата координатна система. За представянето на отсечка, използвайте създадена от вас структура (struct, запис). Функцията да върне дължината на общата им част, ако отсечките лежат на права, успоредна на една от координатните оси или -1 в противен случай.
3. Да се напише функция на С++, която отпечатва в конзолата запълнен правоъгълник с размерност NxM, където N, M и символът за отпечатване на правоъгълника се предават като параметри на функцията. Правоъгълникът да е центриран (приема се, че конзолата има 25 реда и 80 символа на ред).
4. Да се напише функция на С++, която приема като параметри цяло число N и двумерна числова квадратна матрица с размери NxN и връща като резултат сумата на елементите над главния ѝ диагонал.
5. Да се напише функция на С++, която приема като параметри параметри цяло число N и двумерна числова квадратна матрица А с размери NxN и връща като резултат транспонираната матрица на А.
6. Да се напише функция int \* Extract(const int), която връща сортиран масив от най-често срещаните цифри в дадено цяло число в интервал [INT\_MIN, INT\_МАX] и завършва с -1

Пример:

4321 -> [1,2,3,4,-1]

112345 -> [1,-1]

-223441 -> [2,4,-1]

1. Да се напише функция на С++, със сигнатура void Revert(char []), която преобразува подадения като параметър символен низ, като замества всички малки букви от латинската азбука с главни и обратно. Да не се използва вградената библиотека string.
2. Да се напише функция double Solve(const char\*const), която приема символен низ, който представлява просто квадратно уравнение от типа а\*х^2+b\*x+c=0 и връща възможно най-големия корен на уравнението. След пояснение: винаги ще имате корени, a b и c са едноцифрени и различни от 0.
3. Да се напише функция със сигнатура bool CheckRectangular(const Point&, const Point&, const Point&), където Point е структура, която има две член-данни за координати в Декартова координатна система. Функцията да връща true ако трите точки в координатната система образуват правоъгълен триъгълник и false в противен случай.
4. Да се напише функция на С++, със сигнатура char\* CheckStr(char[]), която връща масив, който съдържа всички повтарящи се символи във входящия масив.

например: CheckStr("aif8sLtt8f") = f8t

1. Да се напише функция на С++, със сигнатура int CheckTimeStr(char []), която да проверява дали входния параметър е коректно изписан час във формат HH:MM:SS(HH - ча е във формат 0-24, ММ - минути, SS - секунди).

Функцията да връща -1, при некоректни входни данни.

1. Да се напише функция на С++, със сигнатура int NumCalc(long num, int k, int l), която връща като резултат сумата на k-тата и l-тата цифра на num, или -1 при некоректни входни данни.
2. Да се напише функция на С++, със сигнатура bool CheckDate(char []), която да проверява дали входния параметър е коректна дата във формат DD.MM . Да не се използва вградената библиотека string.
3. Да се напише функция на C++, със сигнатура int DigitPos(long num, int k), която връща като резултат k-тата цифра на числото num или -1, ако такава не съществува. Брои се от ляво на дясно и се започва от 1.
4. Да се напише функция на C++, със сигнатура bool doExist(char\* symbols, char\* word), която проверява дали думата word, се среща в символният низ symbols. Да не се използва вградената библиотека string.
5. Да се напише функция на С++, със сигнатура int AnalyzeArray(int arr[], int l), където l е дължината на масива, която да връща 1, ако масивът може да бъде разделен на две части, с равни суми на елементите им и 0 - в противен слувай. Функцията да връща -1, при невалидни входни данни.

Например: AnalyzeArray({1, 3, 0, 0, 4}) = 1; AnalyzeArray({1, 6, 1, 0, 5, 199}) = 0

1. Да се напише функция на C++, със сигнатура bool doExist(char\* symbols, char\* word), която проверява дали думата word, се среща в символният низ symbols, прочетен наобратно.
2. Да се напише функция на С++, със сигнатура double calculateMatrix(int\*\* matr, int m), която приема като параметър квадратна матрица с размерност m и връща като резултат сумата на елементите които са по двата и диагонала.
3. Да се напише функция на С++, със сигнатура int CheckStr(char []), която проверява, дали символите на входния масив, който трябва да е с дължина поне 2 символа, може да се пренаредят, така че той да стане палиндром. Функцията да връща -1, при невалидни входни данни.

Например:

CheckStr("beblehl") = 1 // "belhleb"

CheckStr("asdfgjk") = 0

CheckStr(NULL) = -1

1. Да се напише функция на С++, със сигнатура long ConvertNum(long&), която преобразува подаденото като параметър число в същото число, но прочетено наобратно.
2. Да се напише функция на С++, със сигнатура char\* GetWord(char str[], int index) (параметърът index е излишен). която връща като резултат най-дългата дума в масива str. За дума се счита всяка последователност от символи, която е отделена отляво и отдясно с един или няколко интервала и/или табулации, а също и началото и края на низа.
3. В даден масив от цели числа, фиксиран се нарича такъв елемент, стойността на който е равна на индекса му. Да се напише функция на С++, със сигнатура bool AnalyzeArr(int arr[], int length), която проверява дали в масива arr съществуват поне два фиксирани елемента. Функцията да връща false също и при некоректни данни.

**Забележка:** За решаване на задачите да не се използват никакви външни библиотеки и функции. Може да реализирате собствени функции, ако сметнете за необходимо. В написаните функции, да няма стандартен вход и изход (cin и/или cout).