

	<i>Fájlnev/könyvtár</i>	<i>Feladat</i>										
01.	<i>ANAGRAMMA.CPP</i>	Készítsen programot, amely két szövegsorról eldönti, hogy egymás anagrammái-e!										
02.	<i>ATLAG.CPP</i>	Olvasson be 4 lebegőpontos számot, számítsa ki az átlagukat, majd az eredményt írja ki <i>e</i> , <i>f</i> és <i>g</i> konverzióval és különböző mezőszélességgel!										
03.	<i>DATUMIDO.CPP</i>	Készítsen programot a C-könyvtár dátum- és időkezelő függvényeinek bemutatására! Használja a <i>setlocale()</i> és az <i>strftime()</i> függvényeket!										
04.	<i>FORDIT.CPP</i>	Olvasson be egy legfeljebb. 5-jegyű egész számot és fordított sorrendben írja vissza a szám jegyeit, az értéktelen nullákat helyettesítse szóközzel!										
05.	<i>HEXDUMP.CPP</i>	Írjon programot, amely a parancssorban megadott állomány tartalmát hexadecimális formában megjeleníti.										
06.	<i>KARIR.CPP</i>	Írjon programot, amely karaktereket olvas, és minden karakter esetén kiírja a karakter bináris és decimális értékét! A program a 'q' karakter leütésére álljon meg!										
07.	<i>MAP.CPP</i>	Készítsen a <i>map</i> konténer felhasználását szemléltető példaprogramot!										
08.	<i>MASOL.CPP</i>	Készítsen programot, amely parancssor-paraméterként kapott állomány (például <i>BE.TXT</i>) tartalmát karakterenként nagybetűre átalakítva egy olyan kimenő állományba másolja, amelynek neve megegyezik a bemenő fájlnevével, a kiterjesztése pedig <i>.OUT</i> lesz (például <i>BE.OUT</i>)!										
09.	<i>MATRIXSTL.CPP</i>	Alkalmazza a szabványos STL <i>vector</i> sablonosztályt kétdimenziós egész tömb (mátrix) létrehozásához és kezeléséhez!										
10.	<i>MONDAT.CPP</i>	Olvasson be egy mondatot, amely ponttal végződik. Írjon programot, amely a beolvasott mondat minden szavát új sorba írja! (A szavak között egy vagy több szóköz áll.)										
11.	<i>NAGYEGESZ</i>	Tervezzünk osztályt nagy egész számok tárolására és használatára, melynek jegyeit <i>vector<char></i> típusú tömbben tároljuk!										
12.	<i>REPIR.CPP</i>	<div>Készítsen programot a repülővel utazó utasok nyilvántartására! Az alábbi adatokat kell beolvasni, és a <i>REP.TXT</i> állományba kiírni:</div> <table><tr><td><i>NEV</i></td><td>max. 20 karakteres sztring</td></tr><tr><td><i>UTLEVELSZAM</i></td><td>egész szám</td></tr><tr><td><i>SZ_DATUM</i></td><td>egész szám</td></tr><tr><td><i>R_DATUM</i></td><td>ee/hh/nn (8 karakter)</td></tr><tr><td><i>KM</i></td><td>egész szám</td></tr></table> <div>Írjon programot, amely billentyűzetről beolvassa az adatokat és kiviszi a <i>REP.TXT</i> nevű állományba! Minden rekordot egy újsor ('\n') karakter zárjon! A mezőket egymástól egy szóköz karakter válassza el!</div>	<i>NEV</i>	max. 20 karakteres sztring	<i>UTLEVELSZAM</i>	egész szám	<i>SZ_DATUM</i>	egész szám	<i>R_DATUM</i>	ee/hh/nn (8 karakter)	<i>KM</i>	egész szám
<i>NEV</i>	max. 20 karakteres sztring											
<i>UTLEVELSZAM</i>	egész szám											
<i>SZ_DATUM</i>	egész szám											
<i>R_DATUM</i>	ee/hh/nn (8 karakter)											
<i>KM</i>	egész szám											

	<i>Fájlnev/könyvtár</i>	<i>Feladat</i>
13.	<i>REPOLV.CPP</i>	Írjon olyan C++ programot, amely beolvassa a <i>REP.TXT</i> állomány tartalmát, az adatokat a memóriában láncolva tárolja, és az alábbi feldolgozást végzi: a) ABC sorrendben kiírja azokat az utazók nevét és útlevél számát, akik 10000 km-nél többet repültek, b) kiírja a legtöbbet repült utazó adatait, c) megjeleníti a legfiatalabb utazó nevét és életkorát, valamint születési évét.
14.	<i>SORREND.CPP</i>	Készítsen olyan C++ programot, amely a <i>BEMENO.TXT</i> nevű szöveges állomány első n sorát abc sorrendbe rendezi és a rendezett sorokat a <i>KIMENO.TXT</i> állományba másolja! Az n értékét a billentyűzetről kérje be a program! Ha a beolvasott adatnál kevesebb sor van a bemenő állományban, akkor a program egy figyelmeztető üzenetet jelenítsen meg! (A rendezéshez használja fel a <i>qsort()</i> könyvtári függvényt!)
15.	<i>STAT.CPP</i>	Írjon C++ programot, amely statisztikát készít arról, hogy egy szöveges fájlban az egyes szavak milyen gyakorisággal fordulnak elő. A statisztikát a szavak ABC sorrendje szerinti rendben írja ki a szabványos kimenetre! Feltételezhető, hogy egy szó nem hosszabb 20 karakternél. Szónak tekinthető minden olyan karaktersor, amelyet tetszőleges számú tagoló karakter (szóköz, '\n', vagy tabulátor) határol. A feldolgozandó állományt jellemző fájlmutatót paraméterként vegye át a függvény! A megoldáshoz definiáljon alkalmas adatstruktúrát! A megírandó függvény a hívó programmal csak a paraméter-listáján keresztül kommunikálhat. Ügyeljen arra, hogy a függvényből való kilépésekor minden dinamikusan foglalt memóriaterületet fel kell szabadítani!
16.	<i>STLMAT.CPP</i>	Készítsen példaprogramok az <i>STL</i> matematikai műveletek használatának bemutatására!
17.	<i>SZOKITALALO1.CPP</i>	Írjon egy elrejtett szót betűnként kitalálni segítő játékprogramot! A megoldáshoz használjon C-stílusú (<i>char *</i>) sztringeket!
18.	<i>SZOKITALALO2.CPP</i>	Írjon egy elrejtett szót betűnként kitalálni segítő játékprogramot! A megoldáshoz használja a C++ <i>string</i> típusát!
19.	<i>VEKTOR.CPP</i>	Készítsen a <i>vector</i> konténer felhasználását szemléltető példaprogramot!