

## Lucrarea de laborator Nr. 8

Structura de date FILE în C: crearea, deschiderea, scrierea, redenumire, ștergere și alte operații cu tipurile de fișiere text & fișiere binare în C.

**Sarcină:** Să se scrie un program în C pentru problemele ce urmează, utilizând funcții elaborate de dvs. Algoritmul funcțiilor va fi programat, reieșind din conținutul problemei.

Nr.	Problema
1.	<p>Să se scrie un program în C care să efectueze următoarele operații:</p> <ul style="list-style-type: none"><li>• Creează un fișier textual (lucru.txt) (în locația de salvare a codului sursă);</li><li>• Deschide fișierul textual cu numele dat, permite introducerea unui șir (array unidimensional cu tipul de date char (un șir), citit de la tastatură), stocând șirul introdus în fișierul de lucru;</li><li>• Identifică și numără, în șirul din fișierul de lucru, prezența spațiilor;</li><li>• Afișează pe ecran informația de output referitoare la câte spații sunt în șirul din fișierul de lucru dat;</li><li>• Redenumeste fișierul de lucru "lucru.txt" în "binarFile.bin";</li><li>• Generează numere random și le salvează într-un vector folosind memorie dinamică;</li><li>• Scrie aceste numere în fișier păstrând și informația existentă în el(în fișier) după care eliberează memoria dinamică(adică șterge vectorul);</li><li>• Citește numerele vectorului din fișier, calculează suma numerelor pare citite și afișează rezultatul în fișierul "output.txt";</li><li>• Șterge fișierul "binarFile.bin".</li></ul> <p>Pentru fiecare dintre aceste sarcini, să se scrie funcții C, apelate în funcția <b>main()</b>.</p>
2.	<p>Să se scrie un program în C care să efectueze următoarele operații:</p> <ul style="list-style-type: none"><li>• Creează un fișier textual (lucru.txt) (în locația de salvare a codului sursă);</li><li>• Deschide fișierul textual cu numele dat, permite introducerea unui șir (array unidimensional cu tipul de date char (un șir), citit de la tastatură), stocând șirul introdus în fișierul de lucru;</li><li>• Identifică și înlocuiește toate spațiile cu caracterul * în șirul din fișierul de lucru &amp; Afișează pe ecran informația de output;</li><li>• Redenumeste fișierul de lucru "lucru.txt" în "binarFile.bin";</li><li>• Generează numere random și le salvează într-un vector folosind memorie dinamică;</li><li>• Scrie aceste numere în fișier păstrând și informația existentă în el(în fișier) după care eliberează memoria dinamică(adică șterge vectorul);</li><li>• Citește numerele vectorului din fișier, calculează media aritmetică a numerelor citite și afișează rezultatul în fișierul "output.txt";</li><li>• Șterge fișierul "binarFile.bin".</li></ul> <p>Pentru fiecare dintre aceste sarcini, să se scrie funcții C, apelate în funcția <b>main()</b>.</p>

3.	<p>Să se scrie un program în C care să efectueze următoarele operații:</p> <ul style="list-style-type: none"> <li>• Creează un fișier textual (lucru.txt) (în locația de salvare a codului sursă);</li> <li>• Deschide fișierul textual cu numele dat, permite introducerea unui șir (array unidimensional cu tipul de date char (un șir), citit de la tastatură), stocând șirul introdus în fișierul de lucru;</li> <li>• Identifică și înlocuiește toate spațiile duble cu un singur spațiu în șirul din fișierul de lucru &amp; Afișează pe ecran informația de output a numerelor de spații găsite și șterse;</li> <li>• Redenumeste fișierul de lucru "lucru.txt" în "binarFile.bin";</li> <li>• Generează numere random și le salvează într-un vector folosind memorie dinamică;</li> <li>• Scrie aceste numere în fișier păstrând și informația existentă în el(în fișier) după care eliberează memoria dinamică(adică șterge vectorul);</li> <li>• Citește numerele vectorului din fișier, calculează produsul numerelor impare citite și afișează rezultatul în fișierul "output.txt";</li> <li>• Șterge fișierul "binarFile.bin".</li> </ul> <p>Pentru fiecare dintre aceste sarcini, să se scrie funcții C, apelate în funcția <b>main()</b>.</p>
4.	<p>Să se scrie un program în C care să efectueze următoarele operații:</p> <ul style="list-style-type: none"> <li>• Creează un fișier textual (lucru.txt) (în locația de salvare a codului sursă);</li> <li>• Deschide fișierul textual cu numele dat, permite introducerea unui șir (array unidimensional cu tipul de date char (un șir), citit de la tastatură), stocând șirul introdus în fișierul de lucru;</li> <li>• Identifică cuvinte sunt în șirul din fișierul de lucru &amp; Afișează pe ecran informația de output a numărului de cuvinte găsite;</li> <li>• Redenumeste fișierul de lucru "lucru.txt" în "binarFile.bin";</li> <li>• Generează numere random și le salvează într-un vector folosind memorie dinamică;</li> <li>• Scrie aceste numere în fișier păstrând și informația existentă în el(în fișier) după care eliberează memoria dinamică(adică șterge vectorul);</li> <li>• Citește numerele vectorului din fișier, calculează produsul numerelor impare citite și afișează rezultatul în fișierul "output.txt";</li> <li>• Șterge fișierul "binarFile.bin".</li> </ul> <p>Pentru fiecare dintre aceste sarcini, să se scrie funcții C, apelate în funcția <b>main()</b>.</p>
5.	<p>Să se scrie un program în C care să efectueze următoarele operații:</p> <ul style="list-style-type: none"> <li>• Creează un fișier textual (lucru.txt) (în locația de salvare a codului sursă);</li> <li>• Deschide fișierul textual cu numele dat, permite introducerea unui șir (array unidimensional cu tipul de date char (un șir), citit de la tastatură), stocând șirul introdus în fișierul de lucru;</li> <li>• Identifică și numără prezența virgulelor în șirul din fișierul de lucru &amp; Afișează pe ecran informația de output a numărului de virgule;</li> <li>• Redenumeste fișierul de lucru "lucru.txt" în "binarFile.bin";</li> <li>• Generează numere random și le salvează într-un vector folosind memorie dinamică;</li> <li>• Scrie aceste numere în fișier păstrând și informația existentă în el(în fișier) după care eliberează memoria dinamică(adică șterge vectorul);</li> <li>• Citește numerele vectorului din fișier, numără câte numere pozitive există printre numerele citite și afișează rezultatul în fișierul "output.txt";</li> <li>• Șterge fișierul "binarFile.bin".</li> </ul> <p>Pentru fiecare dintre aceste sarcini, să se scrie funcții C, apelate în funcția <b>main()</b>.</p>

6.	<p>Să se scrie un program în C care să efectueze următoarele operații:</p> <ul style="list-style-type: none"> <li>• Creează un fișier textual (lucru.txt) (în locația de salvare a codului sursă);</li> <li>• Deschide fișierul textual cu numele dat, permite introducerea unui șir (array unidimensional cu tipul de date char (un șir), citit de la tastatură), stocând șirul introdus în fișierul de lucru;</li> <li>• Identifică și numără câte propoziții enunțiative (care se sfârșesc cu punct) sunt în șirul din fișierul de lucru &amp; Afișează pe ecran informația de output a numărului de propoziții;</li> <li>• Redenumeste fișierul de lucru "lucru.txt" în "binarFile.bin";</li> <li>• Generează numere random și le salvează într-un vector folosind memorie dinamică;</li> <li>• Scrie aceste numere în fișier păstrând și informația existentă în el(în fișier) după care eliberează memoria dinamică(adică șterge vectorul);</li> <li>• Citește numerele vectorului din fișier, numără câte numere negative există printre numerele citite și afișează rezultatul în fișierul "output.txt";</li> <li>• Șterge fișierul "binarFile.bin".</li> </ul> <p>Pentru fiecare dintre aceste sarcini, să se scrie funcții C, apelate în funcția <b>main()</b>.</p>
7.	<p>Să se scrie un program în C care să efectueze următoarele operații:</p> <ul style="list-style-type: none"> <li>• Creează un fișier textual (lucru.txt) (în locația de salvare a codului sursă);</li> <li>• Deschide fișierul textual cu numele dat, permite introducerea unui șir (array unidimensional cu tipul de date char (un șir), citit de la tastatură), stocând șirul introdus în fișierul de lucru;</li> <li>• Identifică și numără câte propoziții interogative (care se sfârșesc cu ?) sunt în șirul din fișierul de lucru &amp; Afișează pe ecran informația de output a numărului de propoziții;</li> <li>• Redenumeste fișierul de lucru "lucru.txt" în "binarFile.bin";</li> <li>• Generează numere random și le salvează într-un vector folosind memorie dinamică;</li> <li>• Scrie aceste numere în fișier păstrând și informația existentă în el(în fișier) după care eliberează memoria dinamică(adică șterge vectorul);</li> <li>• Citește numerele vectorului din fișier, calculează media aritmetică a numerelor negative citite și afișează rezultatul în fișierul "output.txt";</li> <li>• Șterge fișierul "binarFile.bin".</li> </ul> <p>Pentru fiecare dintre aceste sarcini, să se scrie funcții C, apelate în funcția <b>main()</b>.</p>
8.	<p>Să se scrie un program în C care să efectueze următoarele operații:</p> <ul style="list-style-type: none"> <li>• Creează un fișier textual (lucru.txt) (în locația de salvare a codului sursă);</li> <li>• Deschide fișierul textual cu numele dat, permite introducerea unui șir (array unidimensional cu tipul de date char (un șir), citit de la tastatură), stocând șirul introdus în fișierul de lucru;</li> <li>• Identifică și numără câte propoziții exclamative (care se sfârșesc cu !) sunt în șirul din fișierul de lucru &amp; Afișează pe ecran informația de output a numărului de propoziții;</li> <li>• Redenumeste fișierul de lucru "lucru.txt" în "binarFile.bin";</li> <li>• Generează numere random și le salvează într-un vector folosind memorie dinamică;</li> <li>• Scrie aceste numere în fișier păstrând și informația existentă în el(în fișier) după care eliberează memoria dinamică(adică șterge vectorul);</li> <li>• Citește numerele vectorului din fișier, calculează media aritmetică a numerelor pozitive citite și afișează rezultatul în fișierul "output.txt";</li> <li>• Șterge fișierul "binarFile.bin".</li> </ul> <p>Pentru fiecare dintre aceste sarcini, să se scrie funcții C, apelate în funcția <b>main()</b>.</p>

9.	<p>Să se scrie un program în C care să efectueze următoarele operații:</p> <ul style="list-style-type: none"> <li>• Creează un fișier textual (lucru.txt) (în locația de salvare a codului sursă);</li> <li>• Deschide fișierul textual cu numele dat, permite introducerea unui șir (array unidimensional cu tipul de date char (un șir), citit de la tastatură), stocând șirul introdus în fișierul de lucru;</li> <li>• Identifică și înlocuiește vocalele cu * în șirul din fișierul de lucru &amp; Afișează pe ecran informația de output a șirului modificat;</li> <li>• Redenumeste fișierul de lucru "lucru.txt" în "binarFile.bin";</li> <li>• Generează numere random și le salvează într-un vector folosind memorie dinamică;</li> <li>• Scrie aceste numere în fișier păstrând și informația existentă în el(în fișier) după care eliberează memoria dinamică(adică șterge vectorul);</li> <li>• Citește numerele vectorului din fișier, calculează media aritmetică a numerelor pare citite și afișează rezultatul în fișierul "output.txt";</li> <li>• Șterge fișierul "binarFile.bin".</li> </ul> <p>Pentru fiecare dintre aceste sarcini, să se scrie funcții C, apelate în funcția <b>main()</b>.</p>
10.	<p>Să se scrie un program în C care să efectueze următoarele operații:</p> <ul style="list-style-type: none"> <li>• Creează un fișier textual (lucru.txt) (în locația de salvare a codului sursă);</li> <li>• Deschide fișierul textual cu numele dat, permite introducerea unui șir (array unidimensional cu tipul de date char (un șir), citit de la tastatură), stocând șirul introdus în fișierul de lucru;</li> <li>• Identifică și înlocuiește vocalele de pe pozițiile pare cu @ în șirul din fișierul de lucru &amp; Afișează pe ecran informația de output a șirului modificat;</li> <li>• Redenumeste fișierul de lucru "lucru.txt" în "binarFile.bin";</li> <li>• Generează numere random și le salvează într-un vector folosind memorie dinamică;</li> <li>• Scrie aceste numere în fișier păstrând și informația existentă în el(în fișier) după care eliberează memoria dinamică(adică șterge vectorul);</li> <li>• Citește numerele vectorului din fișier, calculează media aritmetică a numerelor impare citite și afișează rezultatul în fișierul "output.txt";</li> <li>• Șterge fișierul "binarFile.bin".</li> </ul> <p>Pentru fiecare dintre aceste sarcini, să se scrie funcții C, apelate în funcția <b>main()</b>.</p>
11.	<p>Să se scrie un program în C care să efectueze următoarele operații:</p> <ul style="list-style-type: none"> <li>• Creează un fișier textual (lucru.txt) (în locația de salvare a codului sursă);</li> <li>• Deschide fișierul textual cu numele dat, permite introducerea unui șir (array unidimensional cu tipul de date char (un șir), citit de la tastatură), stocând șirul introdus în fișierul de lucru;</li> <li>• Identifică și rescrie toate consoanele cu literă mare de pe pozițiile pare în șirul din fișierul de lucru &amp; Afișează pe ecran informația de output a șirului modificat;</li> <li>• Redenumeste fișierul de lucru "lucru.txt" în "binarFile.bin";</li> <li>• Generează numere random și le salvează într-un vector folosind memorie dinamică;</li> <li>• Scrie aceste numere în fișier păstrând și informația existentă în el(în fișier) după care eliberează memoria dinamică(adică șterge vectorul);</li> <li>• Citește numerele vectorului din fișier, calculează media aritmetică a numerelor de pe pozițiile impare citite și afișează rezultatul în fișierul "output.txt";</li> <li>• Șterge fișierul "binarFile.bin".</li> </ul> <p>Pentru fiecare dintre aceste sarcini, să se scrie funcții C, apelate în funcția <b>main()</b>.</p>

12.	<p>Să se scrie un program în C care să efectueze următoarele operații:</p> <ul style="list-style-type: none"> <li>• Creează un fișier textual (lucru.txt) (în locația de salvare a codului sursă);</li> <li>• Deschide fișierul textual cu numele dat, permite introducerea unui șir (array unidimensional cu tipul de date char (un șir), citit de la tastatură), stocând șirul introdus în fișierul de lucru;</li> <li>• Identifică și înlocuiește fiecare caracter numeric cu \$ în șirul din fișierul de lucru &amp; Afîșează pe ecran informația de output a șirului modificat;</li> <li>• Redenumeste fișierul de lucru "lucru.txt" în "binarFile.bin";</li> <li>• Generează numere random și le salvează într-un vector folosind memorie dinamică;</li> <li>• Scrie aceste numere în fișier păstrând și informația existentă în el(în fișier) după care eliberează memoria dinamică(adică șterge vectorul);</li> <li>• Citește numerele vectorului din fișier, calculează media aritmetică a numerelor de pe pozițiile pare citite și afîșează rezultatul în fișierul "output.txt";</li> <li>• Șterge fișierul "binarFile.bin".</li> </ul> <p>Pentru fiecare dintre aceste sarcini, să se scrie funcții C, apelate în funcția <b>main()</b>.</p>
13.	<p>Să se scrie un program în C care să efectueze următoarele operații:</p> <ul style="list-style-type: none"> <li>• Creează un fișier textual (lucru.txt) (în locația de salvare a codului sursă);</li> <li>• Deschide fișierul textual cu numele dat, permite introducerea unui șir (array unidimensional cu tipul de date char (un șir), citit de la tastatură), stocând șirul introdus în fișierul de lucru;</li> <li>• Identifică și rescrie toate vocalele cu literă mare în șirul din fișierul de lucru &amp; Afîșează pe ecran informația de output a șirului modificat;</li> <li>• Redenumeste fișierul de lucru "lucru.txt" în "binarFile.bin";</li> <li>• Generează numere random și le salvează într-un vector folosind memorie dinamică;</li> <li>• Scrie aceste numere în fișier păstrând și informația existentă în el(în fișier) după care eliberează memoria dinamică(adică șterge vectorul);</li> <li>• Citește numerele vectorului din fișier, calculează suma numerelor de pe pozițiile impare citite și afîșează rezultatul în fișierul "output.txt";</li> <li>• Șterge fișierul "binarFile.bin".</li> </ul> <p>Pentru fiecare dintre aceste sarcini, să se scrie funcții C, apelate în funcția <b>main()</b>.</p>
14.	<p>Să se scrie un program în C care să efectueze următoarele operații:</p> <ul style="list-style-type: none"> <li>• Creează un fișier textual (lucru.txt) (în locația de salvare a codului sursă);</li> <li>• Deschide fișierul textual cu numele dat, permite introducerea unui șir (array unidimensional cu tipul de date char (un șir), citit de la tastatură), stocând șirul introdus în fișierul de lucru;</li> <li>• Rescrie toate caracterele din șir cu literă mare din fișierul de lucru, determină lungimea șirului &amp; Afîșează pe ecran informația de output a șirului modificat și lungimea sa;</li> <li>• Redenumeste fișierul de lucru "lucru.txt" în "binarFile.bin";</li> <li>• Generează numere random și le salvează într-un vector folosind memorie dinamică;</li> <li>• Scrie aceste numere în fișier păstrând și informația existentă în el(în fișier) după care eliberează memoria dinamică(adică șterge vectorul);</li> <li>• Citește numerele vectorului din fișier, calculează suma numerelor de pe pozițiile pare citite și afîșează rezultatul în fișierul "output.txt";</li> <li>• Șterge fișierul "binarFile.bin".</li> </ul> <p>Pentru fiecare dintre aceste sarcini, să se scrie funcții C, apelate în funcția <b>main()</b>.</p>

15.	<p>Să se scrie un program în C care să efectueze următoarele operații:</p> <ul style="list-style-type: none"> <li>• Creează un fișier textual (lucru.txt) (în locația de salvare a codului sursă);</li> <li>• Deschide fișierul textual cu numele dat, permite introducerea unui șir (array unidimensional cu tipul de date char (un șir), citit de la tastatură), stocând șirul introdus în fișierul de lucru;</li> <li>• Calculează numărul de apariții a unui caracter introdus de la tastatură în șirul din fișierul de lucru &amp; Afișează pe ecran informația de output;</li> <li>• Redenumeste fișierul de lucru "lucru.txt" în "binarFile.bin";</li> <li>• Generează numere random și le salvează într-un vector folosind memorie dinamică;</li> <li>• Scrie aceste numere în fișier păstrând și informația existentă în el(în fișier) după care eliberează memoria dinamică(adică șterge vectorul);</li> <li>• Citește numerele vectorului din fișier, calculează suma numerelor negative de pe pozițiile pare citite și afișează rezultatul în fișierul "output.txt";</li> <li>• Șterge fișierul "binarFile.bin".</li> </ul> <p>Pentru fiecare dintre aceste sarcini, să se scrie funcții C, apelate în funcția <b>main()</b>.</p>
16.	<p>Să se scrie un program în C care să efectueze următoarele operații:</p> <ul style="list-style-type: none"> <li>• Creează un fișier textual (lucru.txt) (în locația de salvare a codului sursă);</li> <li>• Deschide fișierul textual cu numele dat, permite introducerea unui șir (array unidimensional cu tipul de date char (un șir), citit de la tastatură), stocând șirul introdus în fișierul de lucru;</li> <li>• Rescrie toate caracterele din șir cu literă mică din fișierul de lucru, determină lungimea șirului &amp; Afișează pe ecran informația de output a șirului modificat și lungimea sa;</li> <li>• Redenumeste fișierul de lucru "lucru.txt" în "binarFile.bin";</li> <li>• Generează numere random și le salvează într-un vector folosind memorie dinamică;</li> <li>• Scrie aceste numere în fișier păstrând și informația existentă în el(în fișier) după care eliberează memoria dinamică(adică șterge vectorul);</li> <li>• Citește numerele vectorului din fișier, calculează suma numerelor pozitive de pe pozițiile pare citite și afișează rezultatul în fișierul "output.txt";</li> <li>• Șterge fișierul "binarFile.bin".</li> </ul> <p>Pentru fiecare dintre aceste sarcini, să se scrie funcții C, apelate în funcția <b>main()</b>.</p>
17.	<p>Să se scrie un program în C care să efectueze următoarele operații:</p> <ul style="list-style-type: none"> <li>• Creează un fișier textual (lucru.txt) (în locația de salvare a codului sursă);</li> <li>• Deschide fișierul textual cu numele dat, permite introducerea unui șir (array unidimensional cu tipul de date char (un șir), citit de la tastatură), stocând șirul introdus în fișierul de lucru;</li> <li>• Identifică și calculează numărul sumar de apariții ale caracterelor „a” și „b” în șirul din fișierul de lucru &amp; Afișează pe ecran informația de output;</li> <li>• Redenumeste fișierul de lucru "lucru.txt" în "binarFile.bin";</li> <li>• Generează numere random și le salvează într-un vector folosind memorie dinamică;</li> <li>• Scrie aceste numere în fișier păstrând și informația existentă în el(în fișier) după care eliberează memoria dinamică(adică șterge vectorul);</li> <li>• Citește numerele vectorului din fișier, calculează media aritmetică a numerelor pozitive de pe pozițiile pare citite și afișează rezultatul în fișierul "output.txt";</li> <li>• Șterge fișierul "binarFile.bin".</li> </ul> <p>Pentru fiecare dintre aceste sarcini, să se scrie funcții C, apelate în funcția <b>main()</b>.</p>