**3 – Aplicaţii subprograme fișiere și vectori**

1. Să se implementeze un program C++ care să conțină următoarele funcții :
2. *citeste\_fisier()* – citește din fișierul *“vector1.txt”* un vector de *n* elemente
3. *afisare() –* afișează vectorul citit anterior pe ecran
4. *float media() -* scrie în fișierul text *“media.txt”* media aritmetică a elementelor pare din vector; dacă în vector nu există nici un element par, se va scrie în fișier mesajul “Nu exista elemente pare în vector!”
5. *void afisare\_prime() -*  afișează pe ecran elementele prime din vector; dacă în vector nu există nici un element prim, se va afișa pe ecran mesajul “Nu exista elemente prime în vector!”. *Se va folosi funcția* ***int prim(int x)*** *!*
6. Să se implementeze un program C++ care să conțină următoarele funcții :
7. *citeste\_fisier()* – citește din fișierul *“vector2.txt”* un vector de *n* elemente
8. *afisare() –* afișează vectorul citit anterior pe ecran
9. *void afisare\_sume() -* scrie în fișierul text *“sume.txt”* suma cifrelor corespunzătoare fiecărui element din vector. *Se va folosi funcția* ***int sumacifrelor(int x)*** *!*
10. *void afisare\_invers() -*  afișează pe ecran inversul corespunzător fiecărui element din vector. *Se va folosi funcția* ***int inversul(int x)*** *!*
11. Să se implementeze un program C++ care să conțină următoarele funcții :
12. *citeste\_fisier()* – citește din fișierul *“vector3.txt”* un vector de *n* elemente
13. *afisare() –* afișează vectorul citit anterior pe ecran
14. *void afisare\_palindrom() -* scrie în fișierul text *“palindrom.txt”* ultimul element palindrom din vector. *Se va folosi funcția* ***int palindrom(int x)*** *!* Dacă în vector nu există nici un element palindrom, se va afișa pe ecran mesajul “Nu exista elemente palindrom în vector!”.
15. *int maxim() -*  returnează elementul maxim din vector
16. *void pozitii\_maxim() –* afișează pe ecran pozițiile pe care se găsește valoarea maximă în vector, folosind funcția de la punctul **d.**
17. Să se implementeze un program C++ care să conțină următoarele funcții :
18. *citeste\_fisier()* – citește din fișierul *“vector4.txt”* un vector de *n* elemente
19. *afisare() –* afișează vectorul citit anterior pe ecran
20. *void sortare() -* sortează crescător elementele vectorului
21. *void scrie() -* scrie în fișierul *“sortat.txt”* vectorul sortat crescător.