***Subiectul 1***

Fişierul **atestat.in** conţine două linii. Pe prima linie se găsește un număr natural nenul **n,** (**n<1000**), iar pe cea de-a doua linie un şir de **n** numere naturale, de cel mult **4** cifre fiecare, separate prin câte un spaţiu.

Scrieți un program în limbajul Pascal/C/C++, care citește datele din fișier și care:

1. să afişeze pe ecran, numerele pare din şir, separate prin câte un spaţiu, în ordinea citirii lor. Dacă nu există numere pare se va afișa mesajul **NU EXISTA**;
2. să afişeze pe ecran, pe următoarea linie, numărul de elemente din şir, care au cel puțin doi factori în descompunerea în factori primi;
3. să se scrie în fişierul **atestat.out** toate numerele palindroame din şir. Acestea vor fi afișate în ordine descrescătoare, separate prin câte un spaţiu. Șirul conține cel puțin un număr palindrom. Un număr este *palindrom* dacă numărul citit de la stânga la dreapta este egal cu numărul citit de la dreapta la stânga, de exemplu: **121**, **2332**, **33**.

**Notă:** Programul va conţine cel puţin un subprogram definit de utilizator.

**Exemplu:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date de intrare:** | | | | | | | **Date de ieşire:** | | | | | |
| atestat.in | | | | | | |  | | | | | |
| 6 |  |  |  |  |  | a) | 12 | 14 | 6 |  |  |  |
| 12 | 14 | 77 | 151 | 27 | 6 | b) | 4 |  |  |  |  |  |
|  |  |  |  |  |  |  | Fişierul atestat.out conţine: | | | | | |
|  |  |  |  |  |  | c) | 151 | 77 | 6 |  |  |  |

***Subiectul 2***

Fişierul **atestat.in** conţine două linii. Pe prima linie se găsește un număr natural nenul **n,** (**n<1000**), iar pe cea de-a doua linie un şir de **n** numere naturale, de cel mult **4** cifre fiecare, separate prin câte un spaţiu.

Scrieți un program în limbajul Pascal/C/C++, care citește datele din fișier și care:

1. să afişeze pe ecran, numerele formate din exact două cifre, separate prin câte un spaţiu, în ordinea citirii. Dacă nu există numere de două cifre se va afișa mesajul **NU EXISTA**;
2. să se afişeze pe următoarea linie pe ecran numărul de elemente din șir care au suma divizorilor un număr par;
3. să scrie în fişierul **atestat.out** toate numerele din şir care au cifrele de aceeași paritate, în ordine descrescătoare, separate prin câte un spaţiu. Șirul conține cel puțin un număr care are toate cifrele de aceeași paritate.

**Notă:** Programul va conţine cel puţin un subprogram definit de utilizator.

**Exemplu:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date de intrare:** | | | | | | | **Date de ieşire:** | | | | | |
| atestat.in | | | | | | |  | | | | | |
| 6 |  |  |  |  |  | a) | 13 | 14 | 28 | 15 |  |  |
| 13 | 9 | 7 | 14 | 28 | 15 | b) | 5 |  |  |  |  |  |
|  |  |  |  |  |  | c) | Fişierul atestat.out conţine: | | | | | |
|  |  |  |  |  |  |  | 28 | 15 | 13 | 9 | 7 |  |

***Subiectul 3***

Fişierul **atestat.in** conţine două linii. Pe prima linie se găsește un număr natural nenul **n,** (**n<1000**), iar pe cea de-a doua linie un şir de **n** numere naturale, de cel mult **9** cifre fiecare, separate prin câte un spaţiu.

Scrieți un program în limbajul Pascal/C/C++, care citește datele din fișier și care:

1. să afişeze pe ecran, numerele impare din şir, separate prin câte un spaţiu, în ordinea inversă citirii. Daca nu există numere impare se va afișa mesajul **NU EXISTA**;
2. să se afişeze pe următoarea linie, pe ecran, numărul de elemente din șir care au cifrele în ordine crescătoare (ordinea crescătoare va fi considerată de la stânga la dreapta);
3. să scrie în fişierul **atestat.out** toate numerele din şir care au suma divizorilor proprii mai mare decât numărul, în ordine crescătoare, separate prin câte un spaţiu. Șirul conține cel puțin un număr care are suma divizorilor proprii mai mare decât el.

**Notă:** Programul va conţine cel puţin un subprogram definit de utilizator.

**Exemplu:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date de intrare:** | | | | | | | **Date de ieşire:** | | | | | |
| atestat.in | | | | | | |  | | | | | |
| 6 |  |  |  |  |  | a) | 3465 | 55 | 123 |  |  |  |
| 123 | 55 | 372 | 3465 | 242 | 44 | b) | 3 |  |  |  |  |  |
|  |  |  |  |  |  |  | Fişierul atestat.out conţine: | | | | | |
|  |  |  |  |  |  | c) | 372 | 3465 |  |  |  |  |

***Subiectul 4***

Fişierul **atestat.in** conţine două linii. Pe prima linie se găsește un număr natural nenul **n,** (3<**n<1000**), iar pe cea de-a doua linie un şir de **n** numere naturale, de cel mult **9** cifre fiecare, separate prin câte un spaţiu.

Scrieți un program în limbajul Pascal/C/C++, care citește datele din fișier și care:

1. să afişeze pe ecran, pătratele perfecte, separate prin câte un spaţiu (afișarea se va realiza în ordinea citirii elementelor). Dacă nu există pătrate perfecte, se va afișa mesajul **NU EXISTA**;
2. să se afişeze pe următoarea linie, pe ecran, numărul de elemente din șir care conțin cifra 0;
3. să scrie în fişierul **atestat.out** toate numerele din şir care au suma divizorilor un număr par, în ordine descrescătoare, separate prin câte un spaţiu. Șirul conține cel puțin un număr care are suma divizorilor un număr par.

**Notă:** Programul va conţine cel puţin un subprogram definit de utilizator.

**Exemplu:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date de intrare:** | | | | | | | **Date de ieşire:** | | | | | |
| atestat.in | | | | | | |  | | | | | |
| 6 |  |  |  |  |  | a) | 25 | 64 |  |  |  |  |
| 1233 | 25 | 1785 | 64 | 409 | 457 | b) | 1 |  |  |  |  |  |
|  |  |  |  |  |  |  | Fişierul atestat.out conţine: | | | | | |
|  |  |  |  |  |  | c) | 1785 | 1233 | 457 | 409 |  |  |

***Subiectul 5***

Fişierul **atestat.in** conţine două linii. Pe prima linie se găsește un număr natural nenul **n,** (**n<1000**), iar pe cea de-a doua linie un şir de **n** numere naturale formate din cel puțin două cifre, separate prin câte un spaţiu.

Scrieți un program în limbajul Pascal/C/C++, care citește datele din fișier și care:

1. să afişeze pe ecran, numerele care au cifra unităților egală cu cifra zecilor, separate prin câte un spaţiu, în ordinea citirii. Daca nu există numere care au cele două cifre egale, se va afișa mesajul **NU EXISTA**;
2. să se afişeze pe următoarea linie pe ecran numărul de elemente din șir care au suma divizorilor un număr impar;
3. să scrie în fişierul **atestat.out** în ordine crescătoare, separate prin câte un spaţiu, toate numerele din şir care au cifrele în ordine descrescătoare (de la stânga la dreapta). Șirul conține cel puțin un număr care are cifrele în ordine descrescătoare.

**Notă:** Programul va conţine cel puţin un subprogram definit de utilizator.

**Exemplu:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date de intrare:** | | | | | | | **Date de ieşire:** | | | | | |
| atestat.in | | | | | | |  | | | | | |
| 6 |  |  |  |  |  | a) | 3455 | 33 |  |  |  |  |
| 3455 | 121 | 9 | 33 | 865 | 321 | b) | 2 |  |  |  |  |  |
|  |  |  |  |  |  |  | Fişierul atestat.out conţine: | | | | | |
|  |  |  |  |  |  | c) | 865 | 321 | 33 |  |  |  |

***Subiectul 6***

Fişierul **atestat.in** conţine două linii. Pe prima linie se găsește un număr natural nenul **n,** (**n<1000**), iar pe cea de-a doua linie un şir de **n** numere naturale distincte, de cel mult **4** cifre fiecare, separate prin câte un spaţiu.

Scrieți un program în limbajul Pascal/C/C++, care citește datele din fișier și care:

1. să afişeze pe ecran, numerele formate din exact doua cifre, separate prin câte un spaţiu, în ordinea inversă citirii. Dacă nu există numere de două cifre se va afișa mesajul **NU EXISTA**;
2. să afişeze pe ecran, pe următoarea linie, numărul de numere perfecte din şir; Un număr este perfect dacă este egal cu suma divizorilor săi strict mai mici decât el. (exemplu 6=1+2+3)
3. să afişeze în fișierul **atestat.ou**t, in ordine crescătoare, toate numerele palindrom din șir. Șirul conține cel puțin un număr palindrom. Un număr este *palindrom* dacă numărul citit de la stânga la dreapta este egal cu numărul citit de la dreapta la stânga, de exemplu: **121**, **222**, **3883**.

**Notă:** Programul va conţine cel puţin un subprogram definit de utilizator.

**Exemplu:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date de intrare:** | | | | | | | **Date de ieşire:** | | | | | |
| atestat.in | | | | | | |  | | | | | |
| 6 |  |  |  |  |  | a) | 81 | 11 | 28 |  |  |  |
| 28 | 11 | 81 | 496 | 6 | 3003 | b) | 3 |  |  |  |  |  |
|  |  |  |  |  |  | c) | Fişierul atestat.out conţine: | | | | | |
|  |  |  |  |  |  |  | 6 | 11 | 3003 |  |  |  |

***Subiectul 7***

Fişierul **atestat.in** conţine două linii. Pe prima linie se găsește un număr natural nenul **n,** (**n<1000**), iar pe cea de-a doua linie un şir de **n** numere naturale distincte, de cel puțin două cifre fiecare, separate prin câte un spaţiu.

Scrieți un program în limbajul Pascal/C/C++, care citește datele din fișier și care:

1. să afişeze pe ecran, numerele divizibile cu primul număr din șir (excepție acesta), separate prin câte un spaţiu, în ordinea citirii. Daca nu există numere divizibile cu primul se va afișa mesajul **NU EXISTA**;
2. să afişeze pe ecran, pe următoarea linie, numărul de numere care au prima cifră egală cu ultima cifră;
3. să afişeze în fișierul **atestat.ou**t, in ordine descrescătoare, toate numerele prime din șir. Șirul conține cel puțin un număr prim.

**Notă:** Programul va conţine cel puţin un subprogram definit de utilizator.

**Exemplu:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date de intrare:** | | | | | | | **Date de ieşire:** | | | | | |
| atestat.in | | | | | | |  | | | | | |
| 6 |  |  |  |  |  | a) | 24 | 3600 |  |  |  |  |
| 12 | 24 | 121 | 3600 | 17 | 131 | b) | 2 |  |  |  |  |  |
|  |  |  |  |  |  | c) | Fişierul atestat.out conţine: | | | | | |
|  |  |  |  |  |  |  | 131 | 17 |  |  |  |  |

***Subiectul 8***

Fișierul **atestat.in** conține două linii. Pe prima linie se găsește un număr natural nenul **n,** (**n<1000**), iar pe cea de-a doua linie un şir de **n** numere naturale, de cel puțin **2** cifre fiecare, separate prin câte un spațiu.

Scrieți un program în limbajul Pascal/C/C++, care citește datele din fișier și care:

1. să afișeze pe ecran, elementele din șir divizibile cu ultimul număr din șir (excepție acesta), separate prin câte un spaţiu, în ordinea inversă citirii. Dacă nu există numere divizibile cu ultimul, se va afișa mesajul **NU EXISTA**;
2. să afișeze pe ecran, pe următoarea linie, numărul de numere din șir care se divid la numărul lor de divizori;
3. să scrie în fişierul **atestat.out,** separate prin câte un spaţiu, oglinditele numerelor din şirul dat în ordine descrescătoare.

**Notă**: Programul va conţine cel puţin un subprogram definit de utilizator.

**Exemplu:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date de intrare:** | | | | | | | **Date de ieşire:** | | | | | |
| atestat.in | | | | | | |  | | | | | |
| 6 |  |  |  |  |  | a) | 42 | 84 |  |  |  |  |
| 2552 | 84 | 12 | 42 | 46 | 21 | b) | 2 |  |  |  |  |  |
|  |  |  |  |  |  |  | Fişierul atestat.out conţine: | | | | | |
|  |  |  |  |  |  | c) | 2552 | 64 | 48 | 24 | 21 | 12 |

***Subiectul 9***

Fişierul **atestat.in** conţine două linii. Pe prima linie se găsește un număr natural nenul **n,** (**n<1000**), iar pe cea de-a doua linie un şir de **n** numere naturale, de cel mult **4** cifre fiecare, separate prin câte un spaţiu.

Scrieți un program în limbajul Pascal/C/C++, care citește datele din fișier și care:

1. să afişeze pe ecran, pe prima linie, în ordinea inversă citirii, toate numerele din şir formate din cel puțin două cifre, separate prin câte un spaţiu. Daca nu există numere formate din cel puțin două cifre, se va afișa mesajul **NU EXISTA**;
2. să se tipărească pe următoarea linie, numărul de elemente prime din șir;
3. să scrie în fişierul **atestat.out** în ordine crescătoare, separate prin câte un spațiu, toate numerele din șir care au numărul de cifre pare egal cu numărul de cifre impare. În șir există cel puțin un număr în care numărul de cifre pare este egal cu numărul de cifre impare.

**Notă:** Programul va conţine cel puţin un subprogram definit de utilizator.

**Exemplu:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date de intrare:** | | | | | | | **Date de ieşire:** | | | | | |
| atestat.in | | | | | | |  | | | | | |
| 6 |  |  |  |  |  | a) | 1003 | 242 | 81 | 555 | 13 |  |
| 13 | 555 | 81 | 2 | 242 | 1003 | b) | 2 |  |  |  |  |  |
|  |  |  |  |  |  |  | Fişierul atestat.out conţine: | | | | | |
|  |  |  |  |  |  | c) | 81 | 1003 |  |  |  |  |

***Subiectul 10***

Fişierul **atestat.in** conţine două linii. Pe prima linie se găsește un număr natural nenul **n,** (**n<1000**), iar pe cea de-a doua linie un şir de **n** numere naturale, de cel mult **9** cifre fiecare, separate prin câte un spaţiu.

Scrieți un program în limbajul Pascal/C/C++, care citește datele din fișier și care:

1. să afişeze pe ecran, pe prima linie, în ordinea citirii, separate prin câte un spaţiu, toate numerele din şir formate din cel mult două cifre. Dacă nu există numere formate din cel mult două cifre, se va afișa mesajul **NU EXISTA**.
2. să afişeze pe ecran, pe a doua linie, numărul de elemente din șir care au suma divizorilor pari divizibilă cu numărul de elemente din șir;
3. să scrie în fişierul **atestat.out** în ordine crescătoare, separate prin câte un spațiu, toate numerele care sunt formate doar din cifre identice. În șir există cel puțin un numărul format din cifre identice.

**Notă:** Programul va conţine cel puţin un subprogram definit de utilizator.

**Exemplu:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date de intrare:** | | | | | | | **Date de ieşire:** | | | | | |
| atestat.in | | | | | | |  | | | | | |
| 6 |  |  |  |  |  | a) | 41 | 2 | 11 |  |  |  |
| 1236 | 555 | 41 | 2 | 242 | 11 | b) | 1 |  |  |  |  |  |
|  |  |  |  |  |  |  | Fişierul atestat.out conţine: | | | | | |
|  |  |  |  |  |  | c) | 2 | 11 | 555 |  |  |  |

***Subiectul 11***

Fişierul **atestat.in** conţine pe prima linie numărul **n,** natural nenul (**2<n<20**), ce reprezintă numărul de linii ale unei matrice pătratice **A**. Pe fiecare din următoarele **n** linii, sunt scrise câte **n** numere naturale, formate din exact **2** cifre fiecare, separate prin câte un spaţiu, reprezentând valorile elementelor matricei pătratice **A**. (numerotarea liniilor şi coloanelor matricei începe de la 1)

Scrieți un program în limbajul Pascal/C/C++, care citește datele din fișier și care:

1. să afişeze pe ecran matricea **A**, câte o linie a matricei pe câte o linie a ecranului, elementele fiecărei linii fiind separate prin câte un spaţiu;
2. să afișeze pe următoarea linie, sumele elementelor de pe liniile pare separate prin câte un spațiu;
3. să se scrie în fişierul **atestat.out** separate prin câte un spaţiu, toate numerele prime din matrice. Dacă nu există numere prime se va scrie mesajul **NU EXISTA**

**Notă:** Programul va conţine cel puţin un subprogram definit de utilizator.

**Exemplu:**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date de intrare:** | | | | | | **Date de ieşire:** | | | | | |
| atestat.in | | | | | |  | | | | | |
| 4 |  |  |  |  | a) | 12 |  | 31 |  | 24 | 15 |
| 12 |  | 31 |  | 24 | 15 | 35 |  | 67 |  | 10 | 17 |
| 35 |  | 67 |  | 10 | 17 | 41 |  | 18 |  | 97 | 18 |
| 41 |  | 18 |  | 97 | 18 | 22 |  | 13 |  | 24 | 99 |
| 22 |  | 13 |  | 24 | 99 b) | 129 |  | 158 |  |  |  |
|  |  |  |  |  |  | Fişierul atestat.out conţine: | | | | | |
|  |  |  |  |  | c) | 31 17 97 13 | | | | | |

***Subiectul 12***

Fişierul **atestat.in** conţine pe prima linie numărul **n,** natural nenul (**2<n<20**), ce reprezintă numărul de linii ale unei matrice pătratice **A**. Pe fiecare din următoarele **n** linii, sunt scrise câte **n** numere naturale, formate din cel mult șase cifre fiecare, separate prin câte un spaţiu, reprezentând valorile elementelor matricei pătratice **A**. (numerotarea liniilor şi coloanelor matricei începe de la 1)

Scrieți un program în limbajul Pascal/C/C++, care citește datele din fișier și care:

1. să afişeze pe ecran matricea **A**, câte o linie a matricei pe câte o linie a ecranului, elementele fiecărei linii fiind separate prin câte un spaţiu;
2. să afișeze pe următoarea linie, ultima cifră a produselor elementelor de pe coloanele impare, separate prin câte un spațiu;
3. să scrie în fişierul **atestat.out**, separate prin câte un spaţiu, toate numerele de tip *palindrom* din matricea **A**. Un număr este *palindrom* dacă numărul citit de la stânga la dreapta este egal cu numărul citit de la dreapta la stânga, de exemplu: **121**, **222**, **3883**. Dacă nu există astfel de numere se va scrie pe prima linie a fişierului mesajul **NU EXISTA**

**Notă:** Programul va conţine cel puţin un subprogram definit de utilizator.

**Exemplu:**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date de intrare:** | | | | | | **Date de ieşire:** | | | | | |
| atestat.in | | | | | |  | | | | | |
| 3 |  |  |  |  | a) | 144 |  | 20 |  | 37 |  |
| 144 |  | 20 |  | 37 |  | 13 |  | 212 |  | 44 |  |
| 13 |  | 212 |  | 44 |  | 12 |  | 40 |  | 152 |  |
| 12 |  | 40 |  | 152 |  |  |  |  |  |  |  |
|  |  |  |  |  | b) | 4 6 | | | | | |
|  |  |  |  |  |  | Fişierul atestat.out conţine: | | | | | |
|  |  |  |  |  | c) | 212 44 | | | | | |

***Subiectul 13***

Fişierul **atestat.in** conţine pe prima linie numărul **n,** natural nenul (**2<n<20**), ce reprezintă numărul de linii ale unei matrice pătratice **A**. Pe fiecare din următoarele **n** linii, sunt scrise câte **n** numere naturale, formate din cel mult **4** cifre fiecare, separate prin câte un spaţiu, reprezentând valorile elementelor matricei **A**. (numerotarea liniilor şi coloanelor matricei începe de la 1)

Scrieți un program în limbajul Pascal/C/C++, care citește datele din fișier și care:

1. să afişeze pe ecran matricea **A**, câte o linie a matricei pe câte o linie a ecranului, elementele fiecărei linii fiind separate prin câte un spaţiu;
2. să afișeze pe următoarea linie, sumele elementelor de pe cele două diagonale separate prin câte un spațiu;
3. să scrie în fişierul **atestat.out** separate prin câte un spaţiu, toate numerele *perfecte* din matrice***.*** Un număr este *perfect* dacă este egal cu suma divizorilor lui mai mici decât el.(exemplu 6=1+2+3) Dacă nu există numere perfecte în matrice, se va scrie pe prima linie a fişierului mesajul **NU EXISTA**

**Notă:** Programul va conţine cel puţin un subprogram definit de utilizator.

**Exemplu:**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date de intrare:** | | | | | | **Date de ieşire:** | | | | | |
| atestat.in | | | | | |  | | | | | |
| 3 |  |  |  |  | a) | 12 |  | 31 |  | 24 |  |
| 12 |  | 31 |  | 24 |  | 35 |  | 67 |  | 10 |  |
| 35 |  | 67 |  | 10 |  | 28 |  | 18 |  | 97 |  |
| 28 |  | 18 |  | 97 |  |  |  |  |  |  |  |
|  |  |  |  |  | b) | 176 119 | | | | | |
|  |  |  |  |  |  | Fişierul atestat.out conţine: | | | | | |
|  |  |  |  |  | c) | 28 | | | | | |

***Subiectul 14***

Fişierul **atestat.in** conţine pe prima linie numărul **n,** natural nenul (**2<n<20**), ce reprezintă numărul de linii ale unei matrice pătratice **A**. Pe fiecare din următoarele **n** linii, sunt scrise câte **n** numere naturale, formate din cel mult **9** cifre fiecare, separate prin câte un spaţiu, reprezentând valorile elementelor matricei **A**. (numerotarea liniilor şi coloanelor matricei începe de la 1)

Scrieți un program în limbajul Pascal/C/C++, care citește datele din fișier și care:

1. să afişeze pe ecran matricea **A**, câte o linie a matricei pe câte o linie a ecranului, elementele fiecărei linii fiind separate prin câte un spaţiu;
2. să afișeze pe următoarea linie, valoarea maximă de pe diagonala principală și valoarea minimă de pe diagonala secundară, separate prin câte un spațiu;
3. să scrie în fişierul **atestat.out** separate prin câte un spaţiu, toate *pătratele perfect*e din matricea **A**. Un număr natural **x** este *pătrat perfect* dacă există un alt număr natural **y** astfel încât **x=y2**, de exemplu **16** este pătrat perfect (**42 =16**). Dacă nu există astfel de numere, se va scrie pe prima linie a fişierului mesajul **NU EXISTA**

**Notă:** Programul va conţine cel puţin un subprogram definit de utilizator.

**Exemplu:**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date de intrare:** | | | | | | **Date de ieşire:** | | | | | |
| atestat.in | | | | | |  | | | | | |
| 3 |  |  |  |  | a) | 8 |  | 36 |  | 101 |  |
| 8 |  | 36 |  | 101 |  | 35 |  | 67 |  | 10 |  |
| 35 |  | 67 |  | 10 |  | 5 |  | 311 |  | 9 |  |
| 5 |  | 311 |  | 9 |  |  |  |  |  |  |  |
|  |  |  |  |  | b) | 67 |  | 5 |  |  |  |
|  |  |  |  |  |  | Fişierul atestat.out conţine: | | | | | |
| c) | | | | | | 36 9 | | | | | |

***Subiectul 15***

Fişierul **atestat.in** conţine pe prima linie numărul **n,** natural nenul (**2<n<20**), ce reprezintă numărul de linii ale unei matrice pătratice **A**. Pe fiecare din următoarele **n** linii, sunt scrise câte **n** numere naturale, formate din cel mult **9** cifre fiecare, separate prin câte un spaţiu, reprezentând valorile elementelor matricei **A**. (numerotarea liniilor şi coloanelor matricei începe de la 1)

Scrieți un program în limbajul Pascal/C/C++, care citește datele din fișier și care:

1. să afişeze pe ecran matricea **A**, câte o linie a matricei pe câte o linie a ecranului, elementele fiecărei linii fiind separate prin câte un spaţiu;
2. să afișeze pe următoarea linie, ultima cifră a produsului elementelor nenule aflate pe chenarul matricei;
3. să scrie în fişierul **atestat.out** separate prin câte un spaţiu, toate *numerele care au prima cifră egală cu ultima.* Dacă nu există astfel de numere, se va scrie pe prima linie a fişierului mesajul **NU EXISTA**

**Notă:** Programul va conţine cel puţin un subprogram definit de utilizator.

**Exemplu:**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date de intrare:** | | | | | | **Date de ieşire:** | | | | | |
| atestat.in | | | | | |  | | | | | |
| 3 |  |  |  |  | a) | 1 |  | 311 |  | 234 |  |
| 1 |  | 311 |  | 234 |  | 13 |  | 9 |  | 67 |  |
| 13 |  | 9 |  | 67 |  | 0 |  | 929 |  | 44 |  |
| 0 |  | 929 |  | 44 |  |  |  |  |  |  |  |
|  |  |  |  |  | b) | 4 |  |  |  |  |  |
|  |  |  |  |  | c) | Fişierul atestat.out conţine: | | | | | |
|  |  |  |  |  |  | 1 9 0 929 44 | | | | | |

***Subiectul 16***

Fişierul **atestat.in** conţine pe prima linie numărul **n,** natural nenul (**2<n<20**), ce reprezintă numărul de linii ale unei matrice pătratice **A**. Pe fiecare din următoarele **n** linii, sunt scrise câte **n** numere naturale, formate din cel mult **4** cifre fiecare, separate prin câte un spaţiu, reprezentând valorile elementelor matricei **A**. (numerotarea liniilor şi coloanelor matricei începe de la 1)

Scrieți un program în limbajul Pascal/C/C++, care citește datele din fișier și care:

1. să afişeze pe ecran matricea **A**, câte o linie a matricei pe câte o linie a ecranului, elementele fiecărei linii fiind separate prin câte un spaţiu;
2. să afișeze pe următoarea linie, suma elementelor situate deasupra diagonalei principale și suma elementelor situate sub diagonala principală, separate prin câte un spațiu;
3. să scrie în fişierul **atestat.out** separate prin câte un spaţiu, toate numerele care au cifrele în ordine crescătoare (de la stânga la dreapta). Dacă nu există astfel de numere, se va scrie pe prima linie a fişierului mesajul **NU EXISTA**

**Notă:** Programul va conţine cel puţin un subprogram definit de utilizator.

**Exemplu:**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date de intrare:** | | | | | | **Date de ieşire:** | | | | | |
| atestat.in | | | | | |  | | | | | |
| 3 |  |  |  |  | a) | 15 |  | 266 |  | 9 |  |
| 15 |  | 266 |  | 9 |  | 4 |  | 23 |  | 56 |  |
| 4 |  | 23 |  | 56 |  | 19 |  | 87 |  | 131 |  |
| 19 |  | 87 |  | 131 |  |  |  |  |  |  |  |
|  |  |  |  |  | b) | 331 |  | 110 |  |  |  |
|  |  |  |  |  | c) | Fişierul atestat.out conţine: | | | | | |
|  |  |  |  |  |  | 15 |  | 266 |  | 9 | 4 23 56 19 |

***Subiectul 17***

Fişierul **atestat.in** conţine pe prima linie numărul **n,** natural nenul (**2<n<20**), ce reprezintă numărul de linii şi de coloane ale unei matrice, iar pe următoarele **n** linii câte **n** numere naturale separate prin câte un spaţiu, formate din cel mult **4** cifre fiecare, reprezentând elementele matricei. (numerotarea liniilor şi coloanelor începe de la 1)

Scrieți un program în limbajul Pascal/C/C++, care citește datele din fișier și care:

1. să afişeze pe ecran matricea **A**, câte o linie a matricei pe câte o linie a ecranului, elementele fiecărei linii fiind separate prin câte un spaţiu;
2. să afișeze pe următoarea linie, suma elementelor situate deasupra diagonalei secundare și suma elementelor situate sub diagonala secundară, separate prin câte un spațiu;
3. să scrie în fişierul **atestat.out** separate prin câte un spaţiu, toate numerele care au cifrele în ordine descrescătoare (de la stânga la dreapta). Dacă nu exist ă astfel de numere, se va scrie pe prima linie a fişierului mesajul **NU EXISTA**

**Notă:** Programul va conţine cel puţin un subprogram definit de utilizator.

**Exemplu:**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date de intrare:** | | | | | | **Date de ieşire:** | | | | | |
| atestat.in | | | | | |  | | | | | |
| 3 |  |  |  |  | a) | 129 |  | 124 |  | 946 |  |
| 129 |  | 124 |  | 946 |  | 31 |  | 232 |  | 56 |  |
| 31 |  | 232 |  | 56 |  | 99 |  | 85 |  | 82 |  |
| 99 |  | 85 |  | 82 |  |  |  |  |  |  |  |
|  |  |  |  |  | b) | 284 223 | | | | | |
|  | | | | | | Fişierul atestat.out conţine: | | | | | |
|  |  |  |  |  | c) | 31 99 85 82 | | | | | |

***Subiectul 18***

Fişierul **atestat.in** conţine o singură linie pe care se află scris un text format din cel mult **200** de caractere care conţine cuvinte separate de un spaţiu. Fiecare cuvânt este format din litere mari sau mici ale alfabetului limbii engleze, sau din cifre. Primul şi ultimul caracter ale textului sunt diferite de caracterul spaţiu.

Scrieți un program în limbajul Pascal/C/C++, care citește datele din fișier și care:

1. să afişeze textul dat scris cu litere mari;
2. să afişeze pe următoarea linie a ecranului, textul obținut după ștergerea caracterelor numerice;
3. să scrie în fişierul **atestat.out** textul în care fiecare cuvânt de lungime impară este înlocuit de oglinditul său. Dacă nu există cuvinte de lungime impară, să se tipărească mesajul **NU EXISTĂ.**

**Notă:** Programul va conţine cel puţin un subprogram definit de utilizator.

**Exemplu**:

|  |  |
| --- | --- |
| **Date de intrare:**  atestat.in | Crei45erul 5678 calcu89latorului e2ste microprocesorul |
| **Date de ieşire:**  - pe ecran  - atestat.out | a). CREI45ERUL 5678 CALCU89LATORULUI E2STE MICROPROCESORUL  b). Creierul calculatorului este microprocesorul |
| c). Crei45erul 5678 calcu89latorului ets2e lurosecorporcim |

***Subiectul 19***

Fişierul **atestat.in** conţine o singură linie pe care se află scris un text format din cel mult **200** de caractere care conţine cuvinte separate de un spaţiu. Fiecare cuvânt este format din litere mari sau mici ale alfabetului limbii engleze, sau din cifre. Primul şi ultimul caracter ale textului sunt diferite de caracterul spaţiu.

Scrieți un program în limbajul Pascal/C/C++, care citește datele din fișier și care:

1. să afişeze textul pe o singură linie a ecranului, numai cu litere mici;
2. să afişeze pe următoarea linie a ecranului, textul obținut prin dublarea vocalelor textului;
3. să scrie în fişierul **atestat.out** textul în care fiecare cuvânt care începe cu o vocală este înlocuit de oglinditul său. Dacă nu există cuvinte care încep cu vocală, să se tipărească mesajul **NU EXISTA.**

**Notă:** Programul va conţine cel puţin un subprogram definit de utilizator.

**Exemplu:**

|  |  |
| --- | --- |
| **Date de intrare:**  atestat.in | Se Face sus aerisirea unui COJOC |
| **Date de ieşire:**  - pe ecran  - atestat.out | a) se face sus aerisirea unui cojoc  b) See Faacee suus aaeeriisiireeaa uunuuii COOJOOC  Fişierul atestat.out conţine:  c) Se Face sus aerisirea iunu COJOC |

***Subiectul 20***

Fişierul **atestat.in** conţine o singură linie pe care se află scris un text format din cel mult **200** de caractere care conţine cuvinte separate de unul sau mai multe spații. Fiecare cuvânt este format din litere mari sau mici ale alfabetului limbii engleze, sau din cifre. Primul şi ultimul caracter ale textului sunt diferite de caracterul spaţiu.

Scrieți un program în limbajul Pascal/C/C++, care citește datele din fișier și care:

1. să afişeze textul pe o singură linie a ecranului, fiecare cuvânt începând cu literă mare;
2. să afişeze pe următoarea linie a ecranului, textul obținut prin ștergerea spațiilor inutile (între cuvinte să existe un singur spațiu);
3. să scrie în fişierul **atestat.out** textul în care fiecare cuvânt de lungime pară să fie înlocuit de un șir de “\*” de aceeași lungime cu acesta. Dacă nu există cuvinte de lungime pară, să se tipărească mesajul **NU EXISTA.**

**Notă:** Programul va conţine cel puţin un subprogram definit de utilizator.

**Exemplu:**

|  |  |
| --- | --- |
| **Date de intrare:**  atestat.in | MARIA ARE UN MAR rosu |
| **Date de ieşire:**  - pe ecran  - atestat.out | a) Maria Are Un Mar Rosu  b) MARIA ARE UN MAR rosu  Fişierul atestat.out conţine:  c) MARIA ARE \*\* MAR \*\*\*\* |

Probleme propuse de către:

* + - 1. prof. dr. Anton Carmen Ana – Colegiul Naţional „Gheorghe Şincai” Baia Mare
      2. prof. Belbe Maria – Colegiul Naţional „Gheorghe Şincai” Baia Mare