SUBIECTELE PROBEI PRACTICE PENTRU

EXAMENUL DE ATESTAT PROFESIONAL LA INFORMATICĂ,

2019-2020

PROGRAMARE

SPECIALIZAREA MATEMATICĂ – INFORMATICĂ

Subiectul nr. 1

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, separate prin câte un spaţiu, reprezentând valorile elementelor matricei **A**.

Să se scrie un program, în limbajul Pascal/C/C++, care:

1. scrie pe ecran, pe aceeași linie, elementele de pe diagonala secundară a

matricei **A**, separate prin câte un spaţiu;

1. prin intermediul unui subprogram scrie pe următoarea linie a ecranului,

numărul elementelor impare de sub diagonala principală a matricei **A**;

1. scrie în fişierul **atestat.out**, pe prima linie, separate prin câte un spaţiu,

toate numerele din matricea **A** care au suma cifrelor un număr prim. Dacă nu

există astfel de numere se va scrie pe prima linie a fişierului mesajul **“NU**

**EXISTA ASTFEL DE NUMERE”.**

**Exemplu:**

|  |  |  |
| --- | --- | --- |
| **atestat.in** | **Date de ieşire:** | |
| **4**  **456 345 5832 1024**  **343 25 567 33**  **24 765 12 121**  **43 678 812 1728** | a)  b)  c) | **1024 567 765 43**  3  Fişierul **atestat.out** conţine:  **1024 25 12 43 812** |

Subiectul nr. 2

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, separate prin câte un spaţiu, reprezentând valorile elementelor matricei **A**.

Să se scrie un program, în limbajul Pascal/C/C++, care:

1. afişează pe ecran, în linie, separate prin câte un spaţiu, elementele de pe ultima linie a matricei **A**;
2. prin intermediul unui subprogram afişează pe următoarea linie a ecranului numărul elementelor impare situate deasupra diagonalei principale a matricei;
3. înlocuiește fiecare element din matricea **A** cu numărul său de divizori și scrie în fişierul **atestat.out** matricea nou formată.

**Exemplu:**

|  |  |  |
| --- | --- | --- |
| **atestat.in** | Date de ieşire: | |
| **4**  **45 13 21 1024**  **7 5 56 33**  **12 765 2 89**  **431 67 12 50** | a)  b)  c) | **431 67 12 50**  **4**  Fişierul **atestat.out** conţine:  **6 2 4 11**  **2 2 8 4**  **6 12 2 2**  **2 2 6 6** |

Subiectul nr. 3

Fişierul **atestat.în** conţine două linii. Pe prima linie este scris un număr natural nenul **n, (5<n<30).** Pe cea de-a două linie a fişierului sunt scrise **n** numere naturale separate prin câte un spaţiu, reprezentând un şir de numere formate fiecare din cel mult **4** cifre.

Să se scrie un program, în limbajul Pascal/C/C++, care:

1. afişează pe ecran, în linie, separate printr-un spaţiu, suma divizorilor pozitivi (proprii și improprii ) pentru fiecare număr ;
2. prin intermediul unui subprogram, afişează pe ecran, mesajul **„DA”** dacă numerele sunt în ordine descrescătoare sau **„NU”** în caz contrar.
3. scrie în fişierul **atestat.out** numerele din şir care au proprietatea că sunt dublul elementului următor din șir. Numerele care indeplinesc condiția vor fi afișate câte doua pe linie, cu spațiu, pe linii diferite. Dacă nu există astfel de numere se va scrie pe prima linie a fișierului mesajul **“NU EXISTĂ ASTFEL DE NUMERE”.**

**Exemplu:**

|  |  |  |
| --- | --- | --- |
| **atestat.in** | **Date de ieşire:** | |
| 7  139 70 35 27 242 121 23 | a)  b)  c) | **140 144 48 40 399 133 24**  **NU**  **Fişierul atestat.out conţine:**   1. **70 35**   **242 121** |

Subiectul nr. 4

Fişierul **atestat.în** conţine două linii. Pe prima linie este scris un număr natural nenul **n,(5<n<30).** Pe cea de-a două linie a fişierului sunt scrise **n** numere naturale distincte separate prin câte un spaţiu, reprezentând un şir de numere formate fiecare din cel mult **4** cifre. Cel puțin unul din cele **n** numere din șir este număr imparși ocupă o poziție pară.

Să se scrie un program, în limbajul Pascal/C/C++, care:

1. afişează pe ecran, în linie, separate printr-un spaţiu, toate numerele impare din șir aflate pe poziții pare;
2. prin intermediul unui subprogram, afişează pe ecran, separate prin câte un spațiu, elementele șirului după interschimbarea elementului minim cu cel maxim.
3. scrie în fişierul **atestat.out**, pe o linie, separate prin câte un spațiu, cifra de control a fiecărui număr din șirul inițial: **Cifra de control** se obţine calculcând suma cifrelor numărului, apoi suma cifrelor sumei și tot așa până la obținerea unei singure cifre.

**Exemplu:**

|  |  |  |
| --- | --- | --- |
| **atestat.in** | Date de ieşire: | |
| 7  248 326 27 165 7231 546 190 | a)  b)  c) | **165**  248 326 7231 165 27 546 190  **Fişierul atestat.out conţine:**  5 2 9 3 4 6 1 |

Subiectul nr. 5

Fişierul **atestat.în** conţine două linii. Pe prima linie este scris un număr natural nenul **n, (5<n<30).** Pe cea de-a două linie a fişierului sunt scrise **n** numere naturale separate prin câte un spaţiu, reprezentând un şir de numere formate fiecare din cel mult **4** cifre. Cel puțin unul din cele **n** numere din șir îndeplinește condiția.

Să se scrie un program, în limbajul Pascal/C/C++, care:

1. afişează pe ecran, în linie, separate printr-un spaţiu, toate numerele cu proprietatea că sunt impare şi divizibile cu 3;
2. prin intermediul unui subprogram, afişează pe ecran suma cifrelor elementului maxim din șir;
3. scrie în fişierul **atestat.out**, pe o linie, separate printr-un spaţiu numerele care au proprietatea că au toate cifrele in ordine strict crescatoare. Dacă nu există astfel de numere se va scrie pe prima linie a fişierului mesajul **“NU EXISTĂ NUMERE CU TOATE CIFRELE IN ORDINE STRICT CRESCATOARE”.**

**Exemplu:**

|  |  |  |
| --- | --- | --- |
| **atestat.in** | Date de ieşire: | |
| **7**  **561 162 35 28 9306 321 33** | a)  b)  c) | **561 321 33**  **18**  **Fişierul atestat.out conţine:**  **35 28** |

**Subiectul nr. 6**

Fișierul **atestat.in** conține pe prima linie două numere naturale nenule: **n** și **c**

**(5<n<100,c<10).** Pe a doua linie sunt scrise **n** numere naturale separate prin câte un spațiu, având fiecare cel mult **8** cifre.

Să se scrie un program, în limbajul Pascal/C/C++, care:

1. scrie pe ecran, în linie, separate printr-un spațiu, toate numerele de pe linia a doua a fișierului, care au ultima cifră (cifra unităților) egală cu c.
2. prin intermediul subprogramului **rast** care inversează (răstoarnă) si returnează un număr având cel mult 8 cifre (ex: 1234 devine 4321), afișează răsturnatele tuturor numerelor impare de pe a doua linie a fișierului.
3. scrie în fișierul **atestat.out,** separate prin câte un spațiu, toate numerele

palindrom din șirul dat, care au exact 3 cifre. Un număr este palindrom dacă citit de la stânga la dreapta sau de la dreapta la stânga are aceiași valoare (**ex.151).** Dacă nu există astfel de numere în șir, se va scrie în fișier mesajul "**NU EXISTĂ NUMERE PALINDROM DE TREI CIFRE**".

**Exemplu**:

|  |  |  |
| --- | --- | --- |
| **atestat.in** | Date de ieşire: | |
| **10 3**  **13 797 25 303 482 12321 72 46 282 31** | a)  b)  c) | 13 303  31 797 52 303 12321 13  Fisieru**l atestat.out** conține:  797 303 282 |

**Subiectul nr.7**

Fișierul **atestat.in** conține cel mult **100** de numere naturale, separate prin câte un spațiu, având fiecare cel mult **8** cifre.

Să se scrie un program, în limbajul Pascal/C/C++, care:

1. scrie pe ecran, în linie, separate prin câte un spațiu, numerele pare din șir, aflate pe poziții impare. Dacă nu există astfel de numere în șir, se va scrie pe ecran mesajul "NU EXISTĂ ".
2. prin intermediul subprogramului **cifre** verifică dacă un număr natural cu cel

mult **8** cifre, primit ca parametru, are prima cifră egală cu ultima, scrie pe ecran, toate numerele din șir care au prima cifră egală cu ultima cifră. Dacă nu există astfel de numere în șir se va afișa mesajul "NU EXISTĂ".

1. scrie în fișierul **atestat.out** media aritmetică dintre cel mai mic și cel mai

mare număr din șir.

**Exemplu:**

|  |  |  |
| --- | --- | --- |
| **atestat.in** | **Date de ieșire:** | |
| **7 11 8 16 5 37 21 4714 64 222** | **a)**  **b)**  **c)** | **8 64**  **7 11 8 5 4714 222**  Fisieru**l atestat.out** conține:  **2359.5** |

**Subiectul nr. 8**

Fișierul **atestat.in** conține pe prima linie numărul natural nenul **n,(5<n<100).** Pe a doua linie sunt scrise **n** numere naturale separate prin câte un spațiu, având fiecare cel mult **8** cifre.

Să se scrie un program, în limbajul Pascal/C/C++, care:

1. afişează cifra care apare cel mai frecvent în scrierea tuturor termenilor șirului;

Dacă sunt mai multe astfel de cifre, se vor scrie toate pe ecran, separate prin virgulă;

1. folosind subprogramul **ndiv**, care primește ca parametru un număr cu cel mult

8 cifre și returnează numărul divizorilor proprii ai acestuia, afișează numărul din șir care are cei mai mulţi divizori proprii. Dacă există mai multe astfel de numere, se va afișa primul dintre ele;

1. cu ajutorul subprogramului definit la b), scrie în fișierul **atestat.out**, pe o linie, separate prin câte un spațiu, toate numerele prime din șir.

**Exemplu:**

|  |  |  |
| --- | --- | --- |
| **atestat.in** | **Date de ieșire:** | |
| **10**  **13 797 25 303 482 12321 72 46 282 31** | **a)**  **b)**  **c)** | **2**  **72**  Fisieru**l atestat.out** conține:  **13 797 31** |

**Subiectul nr.9**

Fișierul **atestat.in** conține pe prima linie un număr natural **n,**(5<n<100), iar pe următoarele linii ale fișierului sunt scrise **n** numere reale, având cel mult 6 zecimale, separate prin câte un spațiu.

Să se scrie un program în limbajul Pascal/C/C++, care:

1. scrie pe ecran, în linie, separate prin câte un spațiu, partea fracționară a tuturor numerelor din șir al căror modul este strict mai mic decât **10** (șirul conține cel puțin două astfel de numere).
2. prin intermediul subprogramului **sort**, care ordonează descrescător un vector, scrie pe ecran, separate printr-un spațiu, toate cele **n** numere ale șirului, sortate descrescător.
3. scrie în fișierul **atestat.out,** cu exact trei zecimale, media aritmetică a numerelor din șirul inițial, al căror modul este strict mai mic decât 10.

**Exemplu:**

|  |  |  |
| --- | --- | --- |
| **atestat.in** | **Date de ieşire:** | |
| **10**  **12.52 1.76 0.12 4.51 18.1 22.2 6.3 11.1 3.35 33.65** | a)  b)  c) | **0.76 0.12 0.51 0.3 0.35**  **33.65 22.2 18.1 12.52 11.1 6.3 4.51 3.35 1.76**  0.12  Fişierul **atestat.out** conţine:  **3.206** |

**Subiectul nr.10**

Fişierul **atestat.in** conţine două linii. Pe prima linie este scris un număr natural nenul **n, (5<n<20).** Pe cea de-a doua linie a fişierului sunt scrise **n** numere naturale separate prin câte un spaţiu, formate fiecare din cel mult **4** cifre, reprezentând un şir de **n** numere naturale. Cel puţin unul din cele **n** numere din şir este divizibil cu **5**.

Să se scrie un program în limbajul Pascal/C/C++, care:

1. afişează pe ecran, în linie, separate prin câte un spaţiu, toate numerele din șir divizibile cu **5**;
2. prin intermediul unui subprogram afişează pe următoarea linie a ecranului, pentru fiecare element din șir, valoarea **1** dacă numărul cifrelor sale pare este egal cu numărul cifrelor sale impare sau valoarea **0** în caz contrar; valorile de **1** sau **0** fiind separate prin câte un spaţiu;
3. scrie în fişierul **atestat.out**, separate prin câte un spațiu, valoarea elementului minim și poziția acestuia; dacă există mai multe numere egale cu minimul atunci se va afișa poziția primului.

**Exemplu:**

|  |  |  |
| --- | --- | --- |
| **atestat.in** | Date de ieşire: | |
| **6**  **1234 45 42 845 2332 101** | a)  b)  c) | **45 845**  **1 1 0 0 1 0**  Fişierul **atestat.out** conţine:  **42 3** |

**Subiectul nr.11**

Fişierul **atestat.în** conţine două linii. Pe prima linie este scris un număr natural nenul **n, (5<n<30).** Pe cea de-a două linie a fişierului sunt scrise **n** numere naturale separate prin câte un spaţiu, reprezentând un şir de numere formate fiecare din cel mult **4** cifre.

Să se scrie un program, în limbajul Pascal/C/C++, care:

1. afişează pe ecran, în linie, separate printr-un spaţiu, toate cifrele de rang **2** din scrierea numerelor; dacă numerele sunt formate dintr-o singură cifră se va afişa valoarea zero;
2. prin intermediul unui subprogram, afişează pe ecran prima poziție pe care se găsește elementul minim în șir;
3. scrie în fişierul **atestat.out**, separate prin câte un spaţiu, toate numerele **prime** care au exact 3 cifre, din șirul dat**.** Un număr natural este **prim**dacă are exact doi divizori: numărul 1 și numărul în sine. Dacă nu există numere

prime cu trei cifre se va scrie pe prima linie a fişierului mesajul **“NU EXISTĂ NUMERE PRIME DE TREI CIFRE”.**

**Exemplu:**

|  |  |  |
| --- | --- | --- |
| **atestat.in** | **Date de ieşire:** | |
| **7**  **239 1541 163 8 1531 19 23** | a)  b)  c) | **3 4 6 0 3 1 2**  **4**  **Fişierul atestat.out conţine:**  **239 163** |

**Subiectul nr.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 **4** cifre, separate prin câte un spaţiu, reprezentând valorile elementelor matricei **A**.

Să se scrie un program, în limbajul Pascal/C/C++, care:

1. afişează pe ecran, în linie, separate prin câte un spaţiu, elementele de pe ultima coloană a matricei **A**;
2. prin intermediul unui subprogram afişează pe următoarea linie a ecranului numărul elementelor pare situate deasupra diagonalei secundare a matricei;
3. înlocuiește fiecare element din matricea **A** cu suma cifrelor sale și scrie în fişierul **atestat.out** matricea nou formată.

**Exemplu:**

|  |  |  |
| --- | --- | --- |
| **atestat.in** | Date de ieşire: | |
| **4**  **12 45 7 991**  **3 46 783 5**  **114 13 21 76**  **65 17 195 1234** | a)  b)  c) | **991 5 76 1234**  **3**  Fişierul **atestat.out** conţine:  **3 9 7 19**  **3 10 18 5**  **6 4 3 13**  **11 8 15 10** |

**Subiectul nr.13**

Fişierul **atestat.in** conţine pe prima linie numerele naturale nenule **m** și **n** **(2<m,n<20),** care reprezintă numărul de linii, respectiv numărul de coloane ale unei matrice **A**. Pe fiecare din următoarele **m** linii, sunt scrise câte **n** numere naturale, formate din cel mult **4** cifre și cel puțin **2** cifre, separate prin câte un spaţiu, reprezentând valorile elementelor matricei **A**.

Să se scrie un program, în limbajul Pascal/C/C++, care:

1. afişează pe ecran, în linie, separate prin câte un spațiu, elementele impare din matricea **A**;
2. prin intermediul unui subprogram afişează pe ecran mesajul **„DA”** dacă un număr natural **k,** de cel mult **4** cifre, introdus de la tastatură, se găsește în matricea **A** sau mesajul **„NU”** în caz contrar;
3. înlocuiește fiecare element din matricea **A** cu produsul cifrelor sale și scrie în fişierul **atestat.out** matricea nou formată.

**Exemplu:**

|  |  |  |
| --- | --- | --- |
| **atestat.in** | Date de ieşire: | |
| **3 4**  **22 13 77 991**  **311 15 784 50**  **9456 1234 428 23**  Se citește de la tastatură **k=15** | a)  b)  c) | **13 77 991 311 15 23**  **DA**  Fişierul **atestat.out** conţine:  **4 3 49 81**  **3 5 224 0**  **1080 24 64 6** |

Subiectul nr. 14

Fişierul **atestat.in** conţine pe prima linie numerele naturale nenule **m** şi **n** **(2<m,n<20),** care reprezintă numărul de linii, respectiv numărul de coloane ale unei matrice **A**. Pe fiecare din următoarele **m** linii, sunt scrise câte **n** numere întregi, formate din cel mult **4** cifre, separate prin câte un spaţiu, reprezentând valorile elementelor matricei **A**.

Să se scrie un program, în limbajul Pascal/C/C++, care:

1. scrie pe ecran, pe aceeași linie, elementele penultimei coloane a matricei **A**, separate prin câte un spaţiu;
2. prin intermediul unui subprogram scrie pe ecran, pe rândul următor numărul elementelor pozitive din matricea **A**;
3. scrie în fişierul **atestat.out**, numărul obţinut prin adunarea cifrei unităților fiecărui număr din matricea **A**.

**Exemplu:**

|  |  |  |
| --- | --- | --- |
| **atestat.in** | **Date de ieşire:** | |
| **3 4**  **-9 234 2 -37**  **567 -88 123 234**  **34 9 -5 67** | **a)**  **b)**  **c)** | **2 123 -5**  **8**  Fişierul **atestat.out** conţine:  **69** |

Subiectul nr. 15

Fişierul **atestat.in** conţine pe prima linie numerele naturale nenule **m** şi **n** **(2<m,n<20),** care reprezintă numărul de linii, respectiv numărul de coloane ale unei matrice **A**. Pe fiecare din următoarele **m** linii, sunt scrise câte **n** numere reale pozitive, formate din cel mult 6 cifre, separate prin câte un spaţiu, reprezentând valorile elementelor matricei **A**.

Să se scrie un program, în limbajul Pascal/C/C++, care:

1. scrie pe ecran, pe aceeași linie, elementele penultimei linii a matricei **A**, separate prin câte un spaţiu; toate elementele vor fi scrise cu **exact două zecimale**.
2. prin intermediul unui subprogram scrie pe ecran media aritmetică a tuturor numerelor din matricea **A cu exact două zecimale**.
3. scrie în fişierul **atestat.out**, pe prima linie, suma dintre partea întreagă a elementului maxim şi partea întreagă a elementului minim din matricea **A**.

**Exemplu:**

|  |  |  |
| --- | --- | --- |
| **atestat.in** | **Date de ieşire:** | |
| **4 3**  **678.92 12.34 44.5678**  **9.1 5.5 32.1**  **34.997 11.2 79.22**  **0.2 94.0 23.3234** | a)  b)  c) | **34.99 11.20 79.22**  **85.49**  Fişierul **atestat.out** conţine:  **678** |

Subiectul nr. 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, separate prin câte un spaţiu, reprezentând valorile elementelor matricei **A**.

Să se scrie un program, în limbajul Pascal/C/C++, care:

1. scrie pe ecran, pe aceeași linie, elementele de pe diagonala principală a matricei **A**;
2. prin intermediul unui subprogram scrie, pe următoarea linie a ecranului, suma elementelor de sub diagonala secundară a matricei **A**;
3. scrie în fişierul **atestat.out**, pe prima linie, separate prin câte un spaţiu, toate numerele ***pătrate perfecte*** din matricea **A**. Un număr natural **a** se numește ***pătrat perfect*** dacă există un alt număr întreg **b** astfel încât **a=b2**, de exemplu **25=52**. Dacă nu există numere pătrate perfecte se va scrie pe prima linie a fişierului mesajul **“NU EXISTA NUMERE PATRATE PERFECTE”.**

**Exemplu:**

|  |  |  |
| --- | --- | --- |
| **atestat.in** | **Date de ieşire:** | |
| **4**  **645 345 21 1024**  **7 25 567 33**  **12 765 2 89**  **43 678 812 7921** | a)  b)  c) | **645 25 2 7921**  9535  Fişierul **atestat.out** conţine:  **1024 25 7921** |

Subiectul nr. 17

Fişierul **atestat.în** conţine două linii. Pe prima linie sunt scrise trei numere naturale nenule, separate prin câte un spațiu: **n,(5<n<30), p≤9, k≤9 (k<p).** Pe cea de-a două linie a fişierului sunt scrise **n** numere naturale separate prin câte un spaţiu, reprezentând un şir de numere formate fiecare din cel mult **4** cifre.

Să se scrie un program în limbajul Pascal/C/C++, care:

1. afişează pe ecran, în linie, separate printr-un spaţiu, toate numerele din șir care împărţite la **p** dau restul **k**;
2. prin intermediul unui subprogram, afişează pe ecran, separate printr-un spațiu, numărul cifrelor pare din scrierea numerelor din șirul dat;
3. verifică dacă toate elementele din șir sunt distincte două câte două şi afișează în fişierul **atestat.out** un mesaj corespunzător: “ **ELEMENTELE SUNT DISTINCTE**” sau “**ELEMENTELE NU SUNT DISTINCTE**”- în caz contrar.

**Exemplu:**

|  |  |  |
| --- | --- | --- |
| **atestat.in** | **Date de ieşire:** | |
| **7 5 3**  **2643 541 97 238 242 33 4682** | a)  b)  c) | **2643 238 33**  **3 1 0 2 3 0 4**  Fişierul **atestat.out** conţine:  **ELEMENTELE SUNT DISTINCTE** |

Subiectul nr.18

Fişierul **atestat.in** conţine o singură linie pe care se află scris un text format din cel mult **70** de caractere, primul și ultimul caracter fiind literă, iar cuvintele din text sunt separate prin unul sau mai multe spaţii. Fiecare cuvânt este format doar din litere mici ale alfabetului limbii engleze.

Să se scrie un program în limbajul Pascal/C/C++, care:

1. afişează textul pe o singură linie a ecranului, transformând prima literă a fiecarui cuvânt în majusculă;
2. prin intermediul unui subprogram, afişează pe ecran pe următoarea linie, cuvintele din textul inițial, care încep și se termină cu o consoană, separate printr-un spațiu;
3. scrie în fişierul **atestat.out**, pe prima linie, numărul cuvintelor din text care sunt formate din exact patru caractere. Dacă nu există astfel de cuvinte, se va scrie pe prima linie a fişierului mesajul **“NU EXISTĂ ASTFEL DE CUVINTE”.**

**Exemplu:**

|  |
| --- |
| **atestat.in** |
| **sara pe deal buciumul suna cu jale** |

|  |
| --- |
| **Date de ieşire:** |
| 1. **Sara Pe Deal Buciumul Suna Cu Jale** 2. **deal buciumul**   Fişierul atestat.out conţine:   1. **6** |

Subiectul nr.19

Fişierul **atestat.in** conţine o singură linie pe care se află scris un text format din cel mult **70** de caractere, primul și ultimul caracter fiind literă, iar cuvintele din text sunt separate prin unul sau mai multe spaţii. Fiecare cuvânt este format doar din litere mari ale alfabetului limbii engleze.

Să se scrie un program, în limbajul Pascal/C/C++, care:

1. afişează textul pe o singură linie a ecranului, transformând ultima litera a fiecarui cuvant în literă mica;
2. prin intermediul unui subprogram, afişează pe ecran pe următoarea linie, textul inițial din care au fost șterse toate vocalele;
3. scrie în fişierul **atestat.out**, pe prima linie, toate cuvintele din textul inițial care au a doua literă vocală**,** separate printr-un spațiu. Dacă nu există astfel de cuvinte, se va scrie pe prima linie a fişierului mesajul **“NU EXISTĂ ASTFEL DE CUVINTE”.**

**Exemplu:**

|  |
| --- |
| **atestat.in** |
| CLOPOTUL VECHI UMPLE CU GLASUL LUI SARA |

Subiectul nr.20

Fişierul **atestat.in** conţine o singură linie pe care se află scris un text format din cel mult **70** de caractere, primul și ultimul caracter fiind literă, iar cuvintele din text sunt separate prin unul sau mai multe spaţii. Fiecare cuvânt este format doar din litere mici ale alfabetului limbii engleze.

Să se scrie un program în limbajul Pascal/C/C++, care:

1. afişează textul pe o singură linie a ecranului, transformând prima și ultima literă, din fiecare cuvânt în majusculă;
2. prin intermediul unui subprogram, afişează pe ecran pe următoarea linie, toate cuvintele din textul iniţial, care încep cu o consoană și se termină cu o vocală, separate printr-un spatiu. ;
3. scrie în fişierul **atestat.out**, pe prima linie, primul cuvânt din care se sterge pe rând câte o consoană, afișându-se cuvintele obținute, separate printr-un spatiu;

**Exemplu:**

|  |
| --- |
| **atestat.in** |
| stelele nasc umezi pe bolta senina |

|  |
| --- |
| **Date de ieşire:** |
| 1. StelelE NasC UmezI PE BoltA SeninA 2. stelele pe bolta senina   Fişierul atestat.out conţine:   1. telele elele eele eee |

Subiectul nr.21

Fişierul **atestat.in** conţine o singură linie pe care se află scris un text format din cel mult **70** de caractere, primul si ultimul caracter fiind literă, iar cuvintele din text sunt separate printr-un singur spaţiu. Fiecare cuvânt este format din litere mari ale alfabetului limbii engleze.

Să se scrie un program în limbajul Pascal/C/C++, care:

1. afişează toate cuvintele din text, pe ecran, câte un cuvânt pe o linie, numai cu litere mici;
2. prin intermediul unui subprogram, afişează textul inţial după ştergerea ultimului caracter din fiecare cuvânt;
3. scrie pe prima linie a fişierului **atestat.out** primul cuvânt de lungime maximă, din textul iniţial;.

**Exemplu:**

|  |
| --- |
| **atestat.in** |
| SI SURAZAND VOM ADORMI SUB INALTUL |

|  |
| --- |
| **Date de ieşire:** |
| 1. si   surazand  vom  adormi  sub  inaltul  **b) S SURAZAN VO** ADORM SU INALTU  Fişierul **atestat.out** conţine:  **c)SURAZAND** |

**Subiectul nr.22**

Fişierul **atestat.in** conţine o singură linie pe care se află scris un text format din cel mult **70** de caractere, primul şi ultimul caracter fiind litere, iar cuvintele din text sunt separate prin unul sau mai multe spaţii. Fiecare cuvânt este format doar din litere ale alfabetului limbii engleze.

Să se scrie un program, în limbajul Pascal/C/C++, care:

1. scrie textul pe ecran, pe o singură linie, transformat astfel: ultima literă a

fiecărui cuvânt, dacă este consoană, va fi scrisă cu literă mare, iar dacă este

vocală, va fi scrisă cu literă mică;

1. prin intermediul unui subprogram, scrie pe ecran pe următoarea linie, textul

inţial după eliminarea spaţiilor în plus ;

1. scrie în fişierului **atestat.out** pe prima linie, separate printr-un spaţiu, toate

cuvintele din fişierul atestat.in care conţin minim patru caractere şi încep şi se

termină cu aceeaşi literă (indiferent dacă este literă mare sau mică) Dacă nu

există astfel de cuvinte, se va scrie pe prima linie a fişierului mesajul **“NU**

**EXISTA ASTFEL DE CUVINTE”.**

**Exemplu:**

|  |
| --- |
| **atestat.in** |
| **ExamenuL penTru ObtinereA atestatuLUI prOfesIonal Este sImplU** |

|  |
| --- |
| **Date de ieşire:** |
| a) **ExamenuL penTru Obtinerea atestatuLUi prOfesIonaL Este sImplu**  b) **ExamenuL penTru ObtinereA atestatuLUI prOfesIonal Este sImplU**  Fişierul **atestat.out** conţine:  **c)Este** |

**Subiectul nr. 23**

Fişierul **atestat.in** conţine pe prima linie numărul natural nenul **n (2<n<30)**reprezentând numărul de sportivi participanţi la proba de contratimp din cadrul Campionatelor de ciclism ale României. Pe următoarele **n** linii sunt scrise câte două numere naturale **m** şi **s** , reprezentând minutele şi secundele timpilor obţinuţi de ciclişti. Cele două numere naturale **m** şi **s** sunt separate printr-un spaţiu.

Utilizând structuri de date tip înregistrare să se scrie un program, în limbajul Pascal/C/C++, care:

1. scrie pe câte o linie a ecranului, cele **n** rezultate măsurate în secunde;
2. prin intermediul unui subprogram, scrie, pe ecran, numărul de sportivi care au obţinut rezultate la mai puţin de un minut de lider;
3. scrie în fişierul **atestat.out**, pe prima linie, numărul intrării în concurs şi timpul obţinut pentru lider şi pe linia următoare, aceleaşi informaţii dar pentru cel mai slab clasat ciclist.

**Exemplu:**

|  |  |  |
| --- | --- | --- |
| **Atestat.in** | Date de ieşire: | |
| **4**  **41 39**  **45 20**  **41 42**  **39 10** | a)  b)  c) | **2499**  **2720**  **2502**  **2350**  **0**  Fişierul **atestat.out** conţine:  **4 39 minute 10 secunde**  **2 45 minute 20 secunde** |

Subiectul nr. 24

Fişierul **atestat.in** conţine pe prima linie numărul natural nenul **n (2<n<50)**reprezentând numărul de sportivi participanţi la Campionatul Naţional de Haltere**.** Pe următoarele **n** linii sunt scrise câte două numere naturale nenule **s** şi **a**, reprezentând greutatea maximă ridicată la stilul **smuls** şi greutatea maximă ridicată la stilul **aruncat** de către fiecare sportiv participant. Cele două numere **s** şi **a** sunt separate printr-un spaţiu.

Utilizând structuri de date tip înregistrare să se scrie un program, în limbajul Pascal/C/C++, care:

1. scrie în linie, pe ecran, separate prin câte un spaţiu, rezultatele sportivilor care au ridicat cel puţin 80 de kilograme la fiecare proba, sub forma  **(s,a) (greutate smulsă, greutate aruncată**);
2. prin intermediul unui subprogram, afişează, pe linii distincte ale ecranului, cele mai bune trei rezultate la categoria total (smuls plus aruncat);
3. scrie în fişierul **atestat.out**, pe câte un rând, numărul intrării în concurs al celui mai bun sportiv pentru categoriile **smuls**, respectiv, **aruncat**.

**Exemplu:**

|  |  |  |
| --- | --- | --- |
| **atestat.in** | Date de ieşire: | |
| **6**  **80 120**  **85 110**  **70 90**  **85 100**  **90 120**  **65 90** | a)  b)  c) | **(80,120) (85,110) (85,100) (90,120) (90,120)**  **(80,120)**  **(85, 110)**  Fişierul **atestat.out** conţine:  **5**  **1 5** |

**Subiectul nr. 25**

Fişierul **atestat.in** conţine pe prima linie numărul natural nenul **n (2<n<10).** Pe fiecare dintre următoarele **n** linii sunt scrise câte două numere întregi **x** şi **y**, reprezentând coordonatele carteziene ale unui punct din planul **xOy**. Cele două numere **x** şi **y** sunt separate printr-un spaţiu.

Utilizând structuri de date tip înregistrare să se scrie un program, în limbajul Pascal/C/C++, care:

1. afişeză punctele din planul **xOy** care nu sunt situate pe prima bisectoare, sub forma **(x,y) (abscisă,ordonată)** câte unul pe linie;
2. prin intermediul unui subprogram scrie pe o linie a ecranului, separate prin câte un spaţiu, sub forma **(x,y)**, toate punctele care au prima coordonată **(abscisa)** un număr par şi sunt situate în primul cadran;
3. scrie în fişierul **atestat.out**, pe linii distincte, coordonatele celui mai îndepartat şi ale celui mai apropiat punct de punctul de intersecţie a axelor, sub forma **(x,y) (abscisă, ordonată)** câte unul pe linie;

**Exemplu:**

|  |  |  |
| --- | --- | --- |
| **Atestat.in** | Date de ieşire: | |
| **5**  **1 3**  **4 4**  **2 5**  **-3 -3**  **6 2** | a)  b)  c) | **(1,3)**  **(2,5)**  **(6,2)**  **(4,4) (2,5) (6,2)**  Fişierul **atestat.out** conţine:  **(6,2)**  **(1,3)** |