

În declararea alăturată, variabila **x** memorează numele unui elev și cele două medii semestriale obținute de acesta la informatică. Scrieți o secvență de instrucțiuni C/C++ în urma executării căreia să se afișeze pe ecran prima literă a numelui și, pe linia următoare, media anuală la informatică a acestui elev. (6p.)

```
struct elev
{
    char nume[30];
    int media1, media2;
} x;
```

**Exemplu:** dacă elevul are numele **Popescu**, iar cele două medii sunt 9, respectiv 10, se afișează pe ecran

P  
9.5

Variabila **p** memorează simultan numărul de vârfuri ale unui poligon (număr natural din intervalul  $[3, 10^2]$ ) și coordonatele vârfurilor acestuia (abscisa și ordonata) în sistemul de coordonate **xOy** (numere reale).

Știind că expresiile C/C++ de mai jos au ca valori numărul de vârfuri ale unui poligon, abscisa, respectiv ordonata primului său vârf, scrieți definiția unei structuri cu eticheta **poligon**, care permite memorarea datelor despre un poligon, și declarați corespunzător variabila **p**.

```
p.numar
p.varf[0].x
p.varf[0].y
```

(6p.)

Se consideră declararea alăturată, în care variabila **t** memorează, pentru fiecare dintre cele 10 triunghiuri, lungimile laturilor.

Scrieți o expresie C/C++ care are valoarea 1 dacă și numai dacă primul triunghi dintre cele menționate este echilateral.

(6p.)

```
struct triunghi {
    int x,y,z;
} t[10];
```

1. În declararea alăturată, în câmpurile **s** și **d** ale structurii sunt memorate extremitatea stângă, respectiv dreaptă ale fiecăruia dintre cele 20 de intervale închise. O expresie C/C++ a cărei valoare este extremitatea stângă a primului interval este: (4p.)

```
struct interval
{
    float s,d;
} v[20];
```

a. **v[0].s**

b. **s.v[0]**

c. **v(0).s**

d. **s(0).v**

Variabila **m** memorează simultan, pentru fiecare dintre cele 20 de mașini oferite spre închiriere, următoarele date: anul fabricației mașinii (număr natural) și tipul de carburant al acesteia (șir de maximum 50 de caractere).

Știind că expresiile C/C++ de mai jos au ca valori anul fabricației celei de a treia mașini, respectiv tipul de carburant al acesteia, scrieți definiția unei structuri cu eticheta **masina**, care permite memorarea datelor despre o mașină, și declarați corespunzător variabila **m**.

```
m[2].an
m[2].carburant
```

(6p.)

Variabila întreagă **an** memorează anul fabricației unei mașini, iar variabila **carburant**, de tip **char**, memorează o literă, în funcție de tipul de carburant folosit în prezent: litera **B** pentru benzină, litera **M** pentru motorină sau litera **G** pentru gaz petrolier lichefiat. Scrieți o secvență de instrucțiuni C/C++ în urma executării căreia se afișează pe ecran anul fabricației mașinii, urmat, pe linia următoare, de mesajul **rezervor clasic**, dacă mașina folosește carburant de tip benzină sau motorină, sau de mesajul **adaptor GPL** dacă mașina folosește carburant de tip gaz petrolier lichefiat. (6p.)

Variabilele **i** și **s** sunt întregi, iar variabila **p** memorează, pentru fiecare dintre cele 20 de zone de parcare ale unui oraș, numărul de locuri închiriate pe parcursul orei curente, precum și prețul practicat de zona respectivă pentru închirierea unui loc pentru o oră.

```
struct parcare
{ int nrLocuriInchiriate;
  int pretOra;
} p[20];
```

Fără a utiliza alte variabile decât cele menționate, scrieți o secvență de instrucțiuni în urma executării căreia variabila **s** să memoreze suma totală obținută în urma închirierii locurilor de parcare din oraș pe parcursul orei curente. (6p.)

1. Variabila **c**, declarată alăturat, memorează codul și prețul unei cărți. Expresia C/C++ a cărei valoare reprezintă dublul prețului cărții respective este: (4p.)

```
struct carte
{ int cod;
  float pret;
} c;
```

- a. **c.pret\*2**                      b. **c[pret]\*2**                      c. **pret^c\*2**                      d. **pret(c)\*2**

Se consideră declararea alăturată, în care variabilele **t1** și **t2** memorează baza și înălțimea a două triunghiuri.

```
struct triunghi
{ int baza, inaltime;
} t1, t2;
```

Scrieți o secvență de instrucțiuni C/C++ care afișează pe ecran mesajul **aceeasi arie**, dacă cele două triunghiuri au arie egală, sau mesajul **arii diferite**, în caz contrar. (6p.)

1. Variabila **c**, declarată alăturat, memorează titlul și prețul unei cărți. Expresia C/C++ a cărei valoare reprezintă prețul cărții respective majorat cu 50% este: (4p.)

```
struct carte
{ char titlu[21];
  float pret;
} c;
```

- a. **c.pret\*3/2**                      b. **pret.c\*3/2**                      c. **c(pret)\*3/2**                      d. **pret[c]\*3/2**

1. În declarațiile alăturate, variabilele **A** și **B** memorează coordonatele câte unui punct în sistemul de coordonate **xOy**. Indicați expresia care are valoarea 1 dacă și numai dacă cele două puncte coincid. (4p.)

```
struct punct
{ int x,y;
} A,B;
```

- a. **A[x]==B[x] && A[y]==B[y]**                      b. **A.x==B.x && A.y==B.y**  
c. **x[A]==x[B] && y[A]==y[B]**                      d. **x.A==x.B && y.A==y.B**

Se consideră declararea alăturată. Scrieți o secvență de instrucțiuni în urma executării căreia să se afișeze pe ecran mesajul **acceptat**, dacă momentul de timp corespunzător variabilei **start** precede momentul de timp **din aceeași oră**, corespunzător variabilei **stop**, sau mesajul **respins** în caz contrar. (6p.)

```
struct timp
{ int minut;
  int secunda;
} start, stop;
```

Se consideră declararea alăturată. Scrieți o secvență de instrucțiuni în urma executării căreia câmpul **mii** al variabilei **x** să memoreze litera **M**, dacă numărul memorat în câmpul **v** al acesteia este strict mai mare decât 999, sau litera **A** altfel. (6p.)

```
struct numar
{ int v;
  char mii;
} x;
```

În declararea alăturată, variabila **m** memorează data cumpărării (luna și anul) și marca unei mașini. Câmpul **luna** are o valoare din intervalul [1,12], iar câmpul **an** are o valoare din intervalul [1900,2200]. Scrieți o expresie care să aibă valoarea 1, dacă mașina a fost cumpărată în una dintre primele două luni ale anului 2014, sau valoarea 0 în caz contrar. (6p.)

```
struct data {
    int luna, an;
};
struct masina {
    struct data cumparare;
    char marca[20];
} m;
```

4. În declararea de mai jos, câmpurile **x** și **y** ale înregistrării reprezintă coordonatele centrului unui cerc, iar câmpul **r** reprezintă raza acestuia. Scrieți expresia C/C++ care are valoarea 1 dacă și numai dacă cercurile corespunzătoare variabilelor **a** și **b** sunt concentrice, iar raza cercului memorat în variabila **a** este dublul razei cercului memorat în variabila **b**. (4p.)

```
struct cerc
{
    float x, y, r;
};
cerc a,b;
```

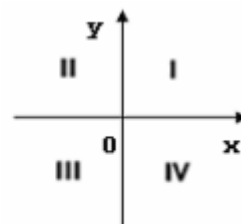
Se consideră declarările alăturate, în care variabila **s** memorează coordonatele, în sistemul de coordonate **xOy**, ale extremităților unui segment.

Scrieți o instrucțiune care afișează pe ecran pătratul lungimii segmentului corespunzător variabilei **s**. (6p.)

```
struct punct {
    float x,y;
};
struct segment {
    punct A,B;
} s;
```

Se consideră declarările alăturate, în care variabila **c** memorează raza și coordonatele, în sistemul de coordonate **xOy**, ale centrului unui cerc.

```
struct punct {
    int x, y;
};
struct cerc {
    struct punct centru;
    float raza;
} c;
```



Scrieți o expresie C/C++ care are valoarea 1 dacă și numai dacă centrul cercului se află în cadranul al II-lea al sistemului de coordonate și nu pe axele acestuia. (6p.)

Se consideră declarările alăturate, în care variabila **s** memorează coordonatele (**x** abscisa, iar **y** ordonata), în sistemul de coordonate **xOy**, ale extremităților unui segment. Scrieți o expresie C/C++ care are valoarea 1 dacă și numai dacă ambele extremități ale segmentului aparțin axei **Ox** a sistemului de coordonate. (6p.)

```
struct punct {
    int x,y;
};
struct segment {
    punct A,B;
} s;
```

Structura cu numele **CARTE** permite memorarea unui număr real, reprezentând valoarea exprimată în euro a unei cărți, a unui număr întreg, reprezentând numărul de pagini ale cărții, și a unui șir de caractere reprezentând titlul cărții, format din cel mult 50 de caractere.

Scrieți în limbajul C/C++ o definiție a structurii menționate și declarați o variabilă **x** de acest tip. Denumiți sugestiv câmpurile acestei structuri. (6p.)

Se consideră tipul **elev**, definit alăturat, ce permite declararea unor variabile care să memoreze anumite date despre un elev: numele și media sa generală anuală.

```
typedef struct {
    char nume[20];
    float medie;
} elev;

int i;
```

Utilizând tipul **elev**, declarați o variabilă, cu numele **clasa**, care să poată memora datele menționate mai sus pentru fiecare dintre cei 30 de elevi ai unei clase, apoi scrieți instrucțiunile care pot completa zona punctată din secvența de instrucțiuni de mai jos, pentru a inițializa, prin citire de la tastatură, variabila respectivă.

```
for (i=0; i<30; i++)
```

.....

(6p.)

1. Se consideră declarațiile alăturate, în care variabila **FIG** memorează coordonatele, în planul **xOy**, ale centrului unui cerc, precum și lungimea razei acestuia. O expresie C/C++ care are valoarea 1 dacă și numai dacă centrul cercului se află pe prima bisectoare a sistemului de coordonate al planului **xOy** este: (4p.)

```
struct punct {
    float x, y;
};
struct cerc {
    struct punct centru;
    float raza;
} FIG;
```

- a. **FIG.centru(x)==FIG.centru(y)**      b. **FIG.centru.x==FIG.centru.y**  
c. **FIG.cerc.x==FIG.cerc.y**      d. **FIG.punct.x==FIG.punct.y**

Variabilele **e1** și **e2**, declarate alăturat, memorează numele și data nașterii pentru câte un elev. Știind că cei doi elevi sunt născuți în ani diferiți, scrieți secvența de instrucțiuni care afișează pe ecran numele elevului de vârstă mai mare. (6p.)

```
struct data_n {
    int an, luna, zi;
};
struct elev{
    char nume[20];
    data_n d;
} e1, e2;
```

3. Se consideră variabila **t**, declarată alăturat, care memorează coordonatele, în planul **xOy**, ale vârfurilor **A**, **B** și **C** ale unui triunghi. Scrieți o instrucțiune care inițializează cu valoarea 0 coordonata **x** a vârfului **A** al triunghiului respectiv. (6p.)

```
struct punct {
    int x, y;
};
struct triunghi {
    struct punct A, B, C;
} t;
```

2. Variabila **s** memorează date specifice despre o seră: numărul de specii de flori (număr natural din intervalul  $[3, 10^2]$ ) și date despre fiecare specie (denumire științifică și denumire populară, șiruri de cel mult 20 de caractere). Expresiile C/C++ de mai jos au ca valori numărul de specii de flori, denumirea științifică, respectiv denumirea populară a celei de a patra specii. Scrieți definiția unei structuri cu eticheta **sera**, care permite memorarea datelor despre o seră, și declarați corespunzător variabila **s**.

**s.numar    s.specie[3].denumireStiintifica    s.specie[3].denumirePopulara** (6p.)

2. Variabila **f** memorează simultan următoarele date despre fiecare dintre cele 20 de specii de flori dintr-o florărie: un cod, reprezentând specia, numărul de varietăți ale speciei respective și numărul de exemplare din fiecare dintre acestea. În florărie sunt maximum 10 varietăți din fiecare specie. Știind că expresiile C/C++ de mai jos au ca valori numere naturale și reprezintă codul și numărul de varietăți ale primei specii, respectiv numărul de exemplare din cea de a 4-a varietate din această specie, scrieți definiția unei structuri cu eticheta **flori**, care permite memorarea datelor despre o specie de flori, și declarați corespunzător variabila **f**.

**f[0].cod    f[0].nrVarietati    f[0].nrExemplare[3]** (6p.)

26 Variabila **t** memorează coordonatele reale (abscisa și ordonata), în planul **xOy**, ale fiecăruia dintre cele trei vârfuri **A**, **B** și **C** ale unui triunghi. Știind că expresiile C/C++ de mai jos au ca valori abscisa vârfului **A** respectiv ordonatele vârfurilor **B** și **C** ale triunghiului, scrieți definiția unei structuri cu eticheta **triunghi**, care permite memorarea datelor precizate, și declarați corespunzător variabila **t**.

**t.A.x    t.B.y    t.C.y**

27. Variabila s memorează simultan următoarele date despre fiecare dintre cele 20 de specii de animale dintr-o rezervație: un cod, reprezentând specia, numărul de exemplare din specia respectivă și vârstele acestora. În rezervație sunt maximum 10 exemplare din fiecare specie. Știind că expresiile C/C++ de mai jos au ca valori numere naturale și reprezintă codul și numărul de exemplare din prima specie, respectiv vârsta celui de al 4-lea exemplar din această specie, scrieți definiția unei structuri cu eticheta specie, care permite memorarea datelor despre o specie, și declarați corespunzător variabila s.

s[0].cod s[0].nrExemplare s[0].varsta[3]

28. Variabila fig memorează date specifice unui cerc: coordonatele reale (abscisa și ordonata), în planul xOy, ale centrului cercului, precum și lungimea razei acestuia. Știind că expresiile C/C++ de mai jos au ca valori numere reale reprezentând datele specifice ale cercului, scrieți definiția unei structuri cu eticheta cerc, care permite memorarea datelor precizate, și declarați corespunzător variabila fig.

fig.centru.x fig.centru.y fig.raza

29. Variabila p memorează date despre un poliedru regulat: numărul de vârfuri, lungimea muchiei și două unghiuri specifice (dintre o față și o muchie, respectiv dintre două fețe). Știind că expresiile C/C++ de mai jos au ca valori un număr natural reprezentând numărul de vârfuri ale poliedrului, și numerele reale reprezentând lungimea muchiei, respectiv cele două unghiuri specifice, scrieți definiția unei structuri cu eticheta poliedru, care permite memorarea datelor precizate, și declarați corespunzător variabila p.

p.NrVarfuri p.Muchie p.Unghi.FataMuchie p.Unghi.FataFata

30. Variabila s memorează simultan numărul de soluții complexe ale unei ecuații (număr natural din intervalul [2,102)) și soluțiile propriu-zise (partea reală și partea imaginară, numere reale). Știind că expresiile C/C++ de mai jos au ca valori numărul de soluții ale unui ecuații, partea reală, respectiv partea imaginară a primei sale soluții, scrieți definiția unei structuri cu eticheta **ecuatie**, care permite memorarea datelor despre soluțiile unei ecuații, și declarați corespunzător variabila s.

s.numar s.solutie[0].pre s.solutie[0].pim

31. Fiecare dintre variabilele A și B, declarate alăturat, memorează coordonatele (x abscisa, iar y ordonata) câte unui punct în sistemul de coordonate xOy. Indicați o expresie C/C++ care are valoarea 1 dacă și numai dacă segmentul cu extremitățile în punctele corespunzătoare variabilelor A și B intersectează axa Oy a sistemului de coordonate.

struct punct

{ int x,y;  
} A,B;

a. A.x\*B.x<=0

b. A(x)\*B(x)<=0

c. x.A\*x.B<=0

d. punct(A,B).y==0

32. Variabila c memorează simultan numărul de cărți dintr-o bibliotecă (număr natural din intervalul [3,102)) și date despre fiecare carte (titlu și autor, șiruri de cel mult 20 de caractere). Expresiile C/C++ de mai jos au ca valori numărul de cărți, titlul și numele autorului celei de a treia cărți. Scrieți definiția unei structuri cu eticheta biblio, care permite memorarea datelor despre o bibliotecă, și declarați corespunzător variabila c.

c.numar

c.carte[2].titlu

c.carte[2].autor

33. Variabila e este declarată alăturat. Indicați o instrucțiune de atribuire corectă din punctul de vedere sintactic.

struct elev

{ int cod;  
char initiala;

struct

{ int nota1,nota2;  
}note;

}e;

a. e(initiala)='A';

b. e=(2020,'A',10,9);

c. e.cod=e.initiala-'A';

d. e.elev.cod=2020;

34. Variabila s memorează simultan următoarele date despre fiecare dintre cele 20 de spectacole din cadrul unei stagiuni: un cod, numărul de actori și vârstele acestora. La un spectacol sunt maximum 10 actori. Expresiile C/C++ de mai jos au ca valori numere naturale și reprezintă codul primului spectacol, numărul de actori care joacă în acest spectacol, respectiv vârsta celui de al 5-lea astfel de actor. Scrieți definiția unei structuri cu eticheta spectacol, care permite memorarea datelor despre un spectacol, și declarați corespunzător variabila s.

s[0].cod                      s[0].nrActori                      s[0].varsta[4]

35. Variabila s memorează date specifice despre o seră: numărul de specii de flori (număr natural din intervalul [3,102)) și date despre fiecare specie (denumire științifică și denumire populară, șiruri de cel mult 20 de caractere). Expresiile C/C++ de mai jos au ca valori numărul de specii de flori, denumirea științifică, respectiv denumirea populară a celei de a patra specii. Scrieți definiția unei structuri cu eticheta sera, care permite memorarea datelor despre o seră, și declarați corespunzător variabila s.

s.numar                      s.specie[3].denumireStiintifica                      s.specie[3].denumirePopulara

36. Variabila c memorează raza și coordonatele (abscisa și ordonata), în planul xOy, ale centrului unui cerc. Știind că expresiile C/C++ de mai jos au valori reale, reprezentând raza, respectiv coordonatele centrului cercului, scrieți definiția unei structuri cu eticheta cerc, care permite memorarea datelor precizate, și declarați corespunzător variabila c.

c.raza                      c.centru.x                      c.centru.y

37. Variabila r memorează simultan următoarele date despre fiecare dintre cele 20 de rase de porumbei dintr-o volieră: un cod, reprezentând rasa, numărul de exemplare din rasa respectivă și vârstele acestora. În volieră sunt maximum 10 exemplare din fiecare rasă. Știind că expresiile C/C++ de mai jos au ca valori numere naturale și reprezintă codul și numărul de exemplare din prima rasă, respectiv vârsta celui de al 4-lea exemplar din această rasă, scrieți definiția unei structuri cu eticheta rasa, care permite memorarea datelor despre o rasă de porumbei, și declarați corespunzător variabila r.

r[0].cod                      r[0].nrExemplare                      r[0].varsta[3]

38. Fiecare dintre variabilele A și B, declarate alăturat, memorează coordonatele pozitive (x abscisa, iar y ordonata) ale câte unui punct în sistemul de coordonate xOy, extremități ale unui segment. Indicați o expresie C/C++ care are valoarea 1 dacă și numai dacă cel puțin una dintre extremitățile segmentului precizat este în originea sistemului de coordonate xOy.

```
struct punct
{ int x,y;
} A,B;
```

- |                                     |   |
|-------------------------------------|---|
| a. $(A.x + A.y) * (B.x + B.y) == 0$ | b. $(A(x) + A(y)) * (B(x) + B(y)) == 0$             |
| c. $(x.A + y.A) * (x.B + y.B) == 0$ | d. $\text{punct}.A(x+y) * \text{punct}.B(x+y) == 0$ |

39. Fiecare dintre variabilele A și B, declarate alăturat, memorează coordonatele (x abscisa, iar y ordonata) câte unui punct în sistemul de coordonate xOy. Indicați o expresie C/C++ care are valoarea 1 dacă și numai dacă segmentul cu extremitățile în punctele corespunzătoare variabilelor A și B intersectează axa Ox a sistemului de coordonate.

```
struct punct
{ int x,y;
} A,B;
```

- |                         |                     |                               |                     |
|-------------------------|---------------------|-------------------------------|---------------------|
| a. $(A-y) * (B-y) <= 0$ | b. $A.y * B.y <= 0$ | c. $\text{punct}.y(A,B) <= 0$ | d. $y.A * y.B <= 0$ |
|-------------------------|---------------------|-------------------------------|---------------------|

40. Variabila ob memorează simultan următoarele date despre un obiectiv turistic din evidența unei agenții: un cod și o perioadă din an în care se recomandă să fie vizitat, formată din zile consecutive, între două date (ziua și luna de început, respectiv ziua și luna finală). Expresiile C/C++ de mai jos au ca valori numere naturale și reprezintă codul obiectivului, ziua și luna de început, respectiv ziua și luna finală a perioadei recomandate pentru vizitarea acestui obiectiv. Scrieți definiția unei structuri cu eticheta obiectiv, care permite memorarea datelor despre un obiectiv turistic, și declarați corespunzător variabila ob.

ob.cod                      ob.dataInceput.zi                      ob.dataInceput.luna                      ob.dataFinal.zi                      ob.dataFinal.luna

41. Variabila p memorează date despre un proiect: tema (un șir de maximum 20 de caractere), numărul surselor de documentare (maximum 10) și, pentru fiecare astfel de sursă, denumirea/adresa web (un șir de maximum 100 de caractere). Știind că expresiile C/C++ de mai jos au ca valori tema unui proiect, numărul de surse de documentare ale acestuia și denumirea/adresa web a primei astfel de surse, scrieți definiția unei structuri cu eticheta proiect, care permite memorarea datelor precizate pentru un proiect, și declarați corespunzător variabila p.

p.tema                      p.nrSurse                      p.denAdrSursa[0]

42. Variabila p memorează simultan date specifice pentru prepararea unei prăjituri: numărul de ingrediente necesare și, pentru fiecare ingredient, codul acestuia și cantitatea necesară. Știind că expresiile C/C++ de mai jos au ca valori numere naturale din intervalul [2,20], reprezentând numărul de ingrediente necesare pentru prepararea unei prăjituri, codul primului ingredient și cantitatea necesară din acesta, scrieți definiția unei structuri cu eticheta prăjitura, care permite memorarea datelor specifice pentru prepararea unei prăjituri, și declarați corespunzător variabila p.

p.numar              p.ingredient[0].cod              p.ingredient[0].cantitate