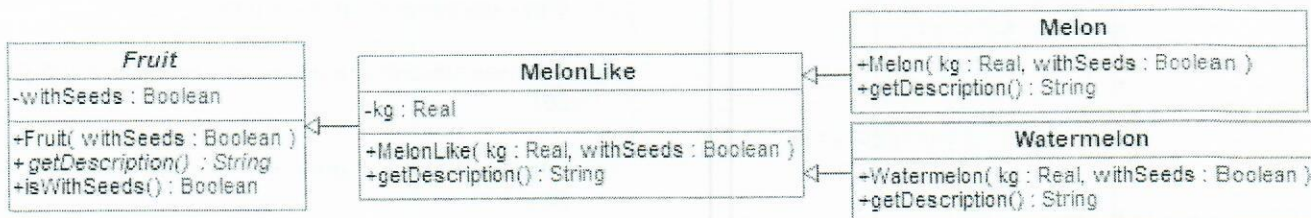


Proba scrisă a examenului de licență, 3 iulie 2017
Informatică Română
VARIANTA 2

SUBIECTUL 1. Algoritmica și programare

Scrieți un program într-unul din limbajele de programare Python, C++, Java, C# care:

a). **Definiște** clasele **Fruit** (fruct), **MelonLike** (pepene), **Melon** (pepene galben) și **Watermelon** (pepene roșu) pe baza următoarei diagrame UML:



- *kg* trebuie să fie o valoare strict pozitivă. Constructorii trebuie să impună constrângerile.
 - Clasa abstractă **Fruit** are o metodă abstractă *getDescription()* (descriere).
 - Metoda *isWithSeeds()* din clasa **Fruit** returnează adevărat dacă fructul e cu sâmburi, fals în caz contrar. Metoda *getDescription()* din **MelonLike** returnează un șir de caractere conținând kilogramele apoi textul “melon like” și apoi textul “with seeds” dacă obiectul are sâmburi sau “without seeds” în caz contrar. Metodele *getDescription()* din **Melon** și **Watermelon** returnează descrierea din clasa de bază concatenată cu textul “melon”, respectiv “watermelon”.
- b). **Definiște o funcție** care returnează poziția pe care ar trebui inserat un **Fruit** într-o listă de **Fruit** ordonată alfabetic după valorile returnate de metoda *getDescription()*. Se va folosi căutare binară.
- c). **Definiște o funcție** care inserează un **Fruit** într-o listă de **Fruit** ordonată alfabetic după valorile returnate de metoda *getDescription()*, folosind funcția de la b).
- d). **Definiște o funcție** care primește ca parametri o valoare booleană *withSeeds* și o listă de **Fruit** și tipărește fructele care au/nu au sâmburi în funcție de valoarea parametrului *withSeeds*.
- e). **Funcția principală** a programului creează o listă cu următoarele fructe: un **Watermelon** fără sâmburi având 6 kg, un **Melon** cu sâmburi având 10 kg, un **MelonLike** fără sâmburi având 11 kg și un **Watermelon** cu sâmburi având 13 kg. Inserați apoi, folosind funcția de la punctul c), un **Watermelon** fără sâmburi având 12 kg în lista construită anterior. Apoi folosiți funcția de la punctul d) pentru a tipări separat fructele cu sâmburi și fructele fără sâmburi.
- f). Pentru tipul de date **Listă** utilizat în program, scrieți specificațiile operațiilor folosite.

Notă

- Se va indica limbajul de programare folosit.
- Nu se vor folosi containere sortate și operații de sortare predefinite.
- Nu se vor defini alte metode decât cele specificate în enunț.

Pentru tipurile de date puteți folosi biblioteci existente (Python, C++, Java, C#).

SUBIECTUL 2. Baze de date

a. Creați o bază de date relațională, având toate tabelele în a treia formă normală, care va reține următoarele informații legate de evenimentul TIFF:

- **locații:** cod locație, denumire, adresă
- **filme:** titlu, an, o listă de genuri (unde un gen are cod, denumire și descriere), o listă de actori (unde fiecare actor are un cod și un nume) și o listă de proiecții ale filmului (unde pentru fiecare proiecție se dă codul proiecției, cod locație, data și ora);

- i. Lista locațiilor (denumire și adresă) unde s-a proiectat cel puțin o comedie și cel puțin o dramă.
- ii. Numărul total de bilete vândute pentru filmele cu actorul *Alain Delon* proiectate în *Piața Unirii*.
- iii. Lista filmelor (titlu, an) pentru care s-au vândut cele mai multe bilete.

SUBIECTUL 3. Sisteme de operare

3.1 Răspundeți la următoarele întrebări, considerând că toate instrucțiunile din fragmentul de cod de mai jos se execută cu succes.

<pre> 1 int main(){ 2 int p[2], i=0; 3 char c, s[20]; 4 pipe(p); 5 if (fork()==0){ 6 close(p[1]); 7 while(read(p[0], &c, sizeof(char))){ 8 if(i < 5 i > 8){ 9 printf("%c", c); 10 } 11 i++; 12 } 13 printf("\n"); close(p[0]); 14 exit(0); 15 } 16 printf("Result: \n"); 17 strcpy(s, "exam not passed"); 18 close(p[0]); 19 write(p[1], s, strlen(s)*sizeof(char)); 20 close(p[1]); 21 wait(NULL); 22 return 0; 23 }</pre>	<ol style="list-style-type: none"> a) Desenați ierarhia proceselor create, incluzând și procesul părinte. b) Dați fiecare linie afișată de program, împreună cu procesul care o tipărește. c) Câte caractere sunt citite din pipe? d) Cum este afectată terminarea proceselor dacă lipsește linia 20? e) Cum este afectată terminarea proceselor dacă lipsesc liniile 20 și 21?
---	--

3.2 Răspundeți la următoarele întrebări, considerând o rulare a scriptului Shell UNIX de mai jos.

<pre> 1 f='find . -type f' 2 d='find . -type d' 3 4 for x in \$f; do 5 for y in \$d; do 6 if [\$x = \$y]; then 7 echo "OK" 8 fi 9 done 10 done</pre>	<ol style="list-style-type: none"> a) De câte ori se afișează „OK”? Justificați răspunsul. b) Care e valoarea variabilei f? c) Care e valoarea variabilei d? d) Care sunt valorile variabilei x? e) Care sunt valorile variabilei y?
--	---

NOTĂ.

- Toate subiectele sunt obligatorii. La toate subiectele se cer rezolvări cu soluții complete.
- Nota minimă ce asigură promovarea este 5,00.
- Timpul efectiv de lucru este de 3 ore.

Subiect 1 (Algoritmica și Programare):

Oficiu – 1p

Definirea claselor *Fruit* și *MelonLike* – 1p din care

relația de moștenire – 0.25

atribute – 0.25

constructor – 0.25

metode - 0.25

Definirea claselor *Watermelon* și *Melon* – 1p din care

relația de moștenire – 0.25

constructor – 0.25

metode – 0.5

Funcția de la punctul b) – 2p din care

signatura corectă - 0.1p

algoritmul de căutare binară - 1.8p

returnare rezultat - 0.1p

Funcția de la punctul c) - 2p din care

signatura corectă - 0.1p

determinare poziție de inserare - cu fct de la b) – 0.2p

inserarea elementului pe poziția determinată anterior – 1.7p

Funcția de la punctul d) – 1p din care

signatura corectă - 0.1p

parcurgere listă – 0.4p

verificare condiție – 0.1p

accesare și tipărire element – 0.4p

Funcția principală e) – 0.5p

f) Specificațiile operațiilor folosite pentru tipul de dată **Listă** – 1.5p**Subiect 2 (Baze de date)**

1 punct oficiu

Problema a:

2 puncte pentru tabelele în 3NF

2 puncte pentru justificare:

1 punct definiții

1 punct explicații

Problema b:

1.5 puncte pentru i

1 punct pentru ii

2.5 puncte pentru iii

Subiect 3 (Sisteme de operare):

Oficiu – 1p

3.1 – 5p din care

a) Diagrama ierarhiei - 1p

b) Linia părintelui – 0.5p

Linia fiului – 0.5p

c) 15 caractere – 1p

d) Niciun proces nu se termină. Fiul blocat la read, părintele la wait – 1p

e) Procesele se termină, pipe-ul fiind închis la terminarea părintelui – 1p

3.2 – 4p din care

a) Nu se afișează nimic – 1p

Justificare – 1p

b) Numele și calea tuturor fișierelor normale din directorul curent și toate subdirectoarele – 0.5p

c) Numele și calea tuturor directoarelor din directorul curent și toate subdirectoarele – 0.5p

d) Numele și calea fiecărui fișier lista din \$f – 0.5p

e) Numele și calea fiecărui director din lista \$d – 0.5p