



Bachelor Degree Exam, September 2015
Computer Science - English

Subject 1

Write a program in one of the programming languages Python, C++, Java, or C# that:

- Defines a class *Medicine*** with a private field *price* of real type, a public constructor that initializes the *price*, and a public method *sellingPrice()* that returns the *price* of the medicine.
- Defines a class *CompensatedMedicine*** derived from *Medicine* with a private field *compensatedPercentage* of real type (representing the percentage from the medicine's price with which it is compensated), a public constructor that initializes the medicine *price* and *compensated percentage*, and also an overridden public method *sellingPrice()* that returns the selling price of the compensated medicine.
- Defines a function** that returns a *medicines* list, containing: an object of type *Medicine* having the *price* 100; an object of type *CompensatedMedicine* having the *price* 70 and the *compensatedPercentage* 0.05; and an object of type *CompensatedMedicine* having the *price* 90 and the *compensatedPercentage* 0.8.
- Defines a function** with parameters a list of type *medicines* like the one from (c) and a real value *v*, and sorts in decreasing order, based on the *selling price*, the medicines from the list having the *selling price* greater than the value *v*. The medicines having the *selling price* less than or equal to *v* will remain on their initial positions in the list.
- Defines a function** with a parameter a list of type *medicines* like the one from (c) and removes from the list the medicines having the *selling price* less than 40.
- Constructs in the **main function** of the program the list indicated at (c), then calls the functions from (d) and then from (e), then prints on the standard output the selling prices of the medicines remained in the list.
- For the *list* data type used in the program write the specifications of the used operations.

Remarks

- **Do not use sorted containers.**
- **Do not define other methods than those required in the subject.**
- **Do not use predefined sorting methods.**

You can use existing libraries for lists (Python, C++, Java, C#). In case you do not use existing libraries, please specify only the operations from the interface of the list.

Subject 2

a. Create a relational database with all tables in 3NF. The database should store the following information about the Untold festival:

- **stages:** stage id, name, address
- **artists:** artist id, name, country of origin, music genre (music genre id, name, description), year of debut, the stage on which the artist performs, the performance day and start time; an artist performs only once during the festival
- **tickets:** ticket code, ticket type (ticket type id, price, name: *early bird*, *full price* or *pay with blood*), the buyer's age, the list of artists at whose concerts the ticket was used to get in.

Justify that the identified tables are in 3NF using functional dependencies.

b. Given the database created at **a**, express the following queries using SQL **OR** the relational algebra:

- b1. The festival's schedule (artist name, concert start time) for 1st of August 2015 on the Cluj Arena stage.
 b2. The stages (name) on which *electro hip hop* music was performed **AND** that hosted concerts with at least one participant over 70 years old.
 b3. The artist (name) with the greatest number of attendants between 18 and 24 years old with *pay with blood* tickets.

Subject 3

3.1 Consider that in the program below all instructions are executed without error and the pipes are closed correctly. Answer the following questions:

<pre> 1 int main() { 2 int f, r, x, p[2]; 3 4 x = getpid(); 5 pipe(p); 6 f = fork(); 7 8 if (x == getpid()) // 1st if 9 close(p[1]); 10 if (f == 0 && x == getpid()) { // 2nd if 11 close(p[0]); 12 write(p[1], &x, sizeof(x)); 13 } 14 if (f > 0) // 3rd if 15 read(p[0], &r, sizeof(r)); 16 if (getppid() == x && f > 0) { // 4th if 17 close(p[1]); 18 read(p[0], &r, sizeof(r)); 19 printf("%d\n", r); 20 exit(0); 21 } 22 if (x == getppid()) { // 5th if 23 close(p[0]); 24 f++; 25 write(p[1], &f, sizeof(f)); 26 exit(0); 27 } 28 if (getppid() == 0) // 6th if 29 printf("%d\n", f); 30 31 printf("%d\n", r); 32 }</pre>	<p>a) What is the meaning of the values stored in x and f variables?</p> <p>b) Explain in detail each if instruction and the corresponding block of code.</p> <p>c) What is printed on standard output when this program is run? Explain why.</p>
---	--

3.2 Consider the UNIX shell script below:

<pre> 1 rm tmp 2 echo -n > tmp 3 for f in \$* 4 do 5 if test ! -f \$f 6 then 7 echo \$f does not exist as a file 8 continue 9 fi 10 rm \$f 11 if [! -f \$f] 12 then 13 echo \$f has been deleted successfully 14 fi 15 ls \$f >> tmp 16 done 17 x=`cat tmp grep -c ^.*\$` 18 echo result: \$x</pre>	<p>a) Explain what is the difference between the if instruction on line 5 and the if instruction on line 11.</p> <p>b) Explain in detail line 17.</p> <p>c) What (and explain why) is the meaning of the value stored in variable x that is printed at the end.</p> <p>d) Rewrite the lines of code from 10 to 14 so that the execution effect remains the same, but using one UNIX command less.</p>
---	--

Remarks: All subjects are compulsory. Each subject will be graded with a mark between 1 and 10 by both evaluators.

Time limit: 3 hours

BAREM INFORMATICĂ

Subiect 1 (Algoritmă și Programare):

Oficiu – 1p

Definierea clasei *Medicament* – 0.75p din care

atribut – 0.25

constructor – 0.25

metoda *prețVânzare()* - 0.25

Definierea clasei *MedicamentCompensat* – 1.75p din care

relația de moștenire – 0.25

constructor – 0.5

atribut – 0.25

metoda *prețVânzare()* – 0.75

Funcția de la punctul c) – 1p din care

signatura corectă și declarare listă - 0.25p

creare obiecte – 0.25p

adăugare obiecte în listă - 0.25p

returnare rezultat - 0.25p

Funcția de la punctul d) – 1.5p din care

signatura corectă - 0.25p

sortare listă conform cerințelor – 1p

returnare rezultat - 0.25p

Funcția de la punctul e) – 1.5p din care

signatura corectă - 0.25p

parcursere listă și ștergere elemente cerute – 1p

returnare rezultat - 0.25p

Program – 1p din care

apel funcții – 0.25p

afișarea prețuri din listă – 0.75p

Specificațiile operațiilor folosite din tipul de dată *Listă* – 1.5p

Subiect 2 (Baze de date):

1 punct oficiu

a) 2p justificare

2p tabele corecte in 3NF

b) b1 - 1p

b2 - 2p

0.5p pentru scenele cu muzica electro hip hop

0.5p pentru scenele care au gazduit concerte cu cel puțin un participant peste 70 ani

1p instructiunea finala

b3- 2p

1p grupare dupa artisti, calculare numar participanti cu conditie

1p instructiunea finala

Subiect 3 (Sisteme de operare):

Oficiu: 1p

3.1

a) 0.5p valoarea variabilei **x**, 0.5p valoarea variabilei **f**

b) 0.5p x 6 **if**-uri

c) Se afișează valoarea 1 + explicație de ce: 1p

3.2

a) nu sunt diferite 1p

b) explicații detaliate 1p

c) numărul de fișiere care nu au putut fi șterse 1p

d) `if rm $f`, se elimină astfel folosirea comenzii `test` sau [1p