Examen de licenţă 2012, specializarea Informatică - model de subiect

1. FP (circa 40'):

Modelaţi problema de mai jos folosind tehnica Backtraking, specificând schematic (precizarea elementelor cheie specifice metodei) construcţia unei soluţii. Creaţi un exemplu pentru care prezentaţi construcţia soluţiei. Justificaţi alegerea tehnicii pentru rezolvarea problemei.

La un concurs sportiv s-au înscris n concurenţi, având numerele de concurs 1, 2,…, n. Pentru fiecare sportiv se cunoaşte ţara de origine (un şir de caractere). In fiecare zi a competiţiei vor intra în concurs m concureni (m <= n). Afisaţi toate posibilităţile de a stabili ordinea intrării în concurs a celor m concurenti, ştiind cã:

a. Doi sportivi din aceeaşi ţară nu pot participa unul după altul.
b. In cadrul unei zile concurenţii intrã în concurs în ordinea numãrului de concurs.

2. OOP (circa 35')

Pentru gestiunea evenimentelor personale, implementati urmatoarele intr-unul dintre limbajele C++, Java sau C#.

- O clasa *Event* ce reprezinta un eveniment personal avand un singur atribut, numele evenimentului (*name*) reprezentat ca sir de caractere, si o singura metoda, *print* ce afiseaza evenimentul (numele lui) pe iesirea standard.
- O clasa *Appointment* derivata din *Event* avand un singur atribut, *startDate* reprezentat de asemenea ca sir de caractere, si o metoda *print* ce suprascrie metoda cu acelasi nume din clasa de baza.
- O clasă *Agenda* ce reprezintă o listă de evenimente, cu o metoda *add* pentru adaugarea unui eveniment şi o metodă *print* pentru tiparirea evenimentelor pe iesirea standard.
- Un program ce creaza o agenda cu evenimentul "Absolvire facultate" si intalnirea ("Licenta, examen scris", "Luni, de la 8"), după care tipareste agenda.

3. SD (circa 25')

Identificaţi tipul abstract de date (`STIVA`, `COADA`, `LISTA`, `ArboreBinar`) potrivit pentru rezolvarea următoarei probleme și scrieți un algoritm pentru rezolvarea acesteia folosind operațiile din interfața tipului abstract de date (făcând abstracție de implementarea concretă a operațiilor).

Se dă un şir de paranteze (rotunde și drepte). Se cere să se verifice dacă parantezele se închid corect. De exemplu: ([]()) se închid corect, ()][() nu se închid corect.

Se vor specifica operațiile din interfața tipului abstract de date (fără implementarea acestora), se vor specifica și implementa subalgoritmii utilizați.

4. BD (circa 25')

A. Se cere o bază de date relaţională, cu tabele în 3NF, ce gestionează următoarele informaţii dintr-o facultate:
   - discipline: cod, denumire, număr credite;
   - studenti: cod, nume, data naşterii, grupa, anul de studiu, specializarea, lista disciplinelor la care a dat examene (inclusiv data examenului şi nota obţinută).

   Justificaţi că tabelele folosite sunt în 3NF.
B. Pentru baza de date de la punctul precedent se cere o instrucţiune de creare a unui tabel în care să apară restricţiile de integritate cheie primară şi cheie externă (străină). Explicaţi semnificaţia acestor restricţii de integritate.
C. Pentru baza de date de la punctul 1, folosind algebra relaţională **sau** instrucţiuni SELECT-SQL, se cer:
   i. Disciplinele pentru care nu există note de promovare (o disciplină este promovată dacă are cel puţin o notă >=5).
   ii. Studenţii (nume, grupa, nr.discipline promovate) ce au promovat peste 5 discipline. O disciplină se numără o singură dată dacă are cel puţin o notă de promovare.

5. SO (circa 25')

A. Prezentaţi pe scurt stările READY, RUN şi WAIT ale unui proces şi evenimentele care cauzează trecerea unui proces dintr-una într-alta.
B. Completaţi programul următor astfel încât procesul părinte să trimită prin PIPE variabila n procesului fiu şi să primească înapoi valoarea ei dublată. Evitaţi ca procesul fiu să tipărească pe ecran sau să devină zombie.

```
1. int main() {
2. int n=1;
3. if(fork() == 0) {
4. }
5. printf("%d\n", n);
6. return 1;
7. }
```

C. Explicaţi funcţionarea scriptului SH de mai jos şi semnificaţia rezultatului tipărit pe ecran. Modificaţi scriptul astfel încât rezultatul tipărit pe ecran să se aplice doar fişierelor cu mai mult de 10 linii.

```
1. N=0
2. for F in *.txt; do
3. K=`wc -l $F|cut -d" " -f1`
4. N=`expr $N + $K`
5. done
6. echo $N
```

Barem 1. FP:

- 1p oficiu
- 2p justificarea aplicabilităţii tehnicii backtracking şi identificarea elementelor cheie specifice metodei
- 1p repreprezentarea unei soluţii
- 1p identificarea spaţiului soluţiilor
- 2p descrierea modalităţii de generare a unei soluţii
- 2p descrierea condiţiei pentru un candidat valid (care poate conduce la o soluţie)
- 1p descrierea soluţiei (condiţia finală)

Barem 2. OOP:

- Oficiu - 1p
- Event - 2p
- Appointment - 2p
- Agenda - 3p
- Program - 2p

Barem 3. SD:

- Oficiu 1p
- Identificarea Tipului Abstract de Date 1p
- Specificarea operațiilor folosite din TAD 2p
- Specificarea subalgoritmilor din algoritm 1p
- Scrierea subalgoritmilor folosiți în algoritm
- (implementarea subprogramelor folosite în aplicație) 3p
- Algoritmul/programul principal 1p
- Stil 1p

Barem 4. BD:

- 1 punct din oficiu
- Problema A: 1 punct pentru tabele în 3NF; 1 punct pentru justificare
- Problema B: 1 punct pentru crearea tabelului; 1 punct pentru explicatie
- Problema C: 2 puncte pentru C.i; 3 puncte pentru C.ii

Barem 5. SO:

- Oficiu 1p
- A 3p
- B 3p
- c Explicare si functionare 1p
- C Modificare (conditionarea executiei cut cu un if la linia 3) 2p