

PROF. UNIV. DR. ING. AUREL ȘERB

SOFTWARE DE SISTEM ȘI SOFTWARE DE APLICAȚII



Copyright © 2012, **Editura Pro Universitaria**

Toate drepturile asupra prezentei ediții aparțin
Editurii Pro Universitaria

Nicio parte din acest volum nu poate fi copiată fără acordul scris al
Editurii Pro Universitaria

Descrierea CIP a Bibliotecii Naționale a României

ȘERB, AUREL

Software de sistem și software de aplicații /

Aurel Șerb. - București : Pro Universitaria, 2012

Bibliogr.

ISBN 978-606-647-285-2

004(075.8)

PREFAȚĂ

Software-ul calculatoarelor este un domeniu extrem de vast și de dinamic. Pentru studenții specializării “Contabilitate și informatică de gestiune”, care își doresc să devină niște utilizatori mai mult decât rafinați ai calculatoarelor și informaticii, cunoștințele referitoare la software-ul de sistem și software-ul de aplicații sunt unele fundamentale. Realitatea cotidiană arată însă faptul că în fiecare moment apar noi aplicații, sistemele de operare se schimbă, sau sunt perfecționate continuu, apar noi limbaje de programare sau instrumente cu care să fie create noi aplicații, etc. Software-ul de sistem și software-ul de aplicații sunt domenii extrem de vaste, care însumează cunoștințe diferite, din multele discipline, care aparțin limbajelor formale sau limbajelor de programare, fie structurilor de date și algoritmilor, fie translațoarelor și compilatoarelor, fie sistemelor de operare, fie sistemelor informatice, fie claselor de aplicații generare, sau dedicate unor domenii de specializare îngustă, și enumerările ar putea continua cu multe altele. A scrie o carte despre “Software de sistem și software de aplicații” este, așadar, o încercare dificilă, prin care orice autor care își propune să facă o sinteză, în final, va expune cititorului un punct de vedere, poate subiectiv. O carte care să trateze toate problemele referitoare la aceste subiecte este dificil de scris într-un număr limitat de pagini. Cartea aceasta încearcă și să prezinte un punct de vedere echilibrat despre software-ul de sistem și software-ul de aplicații, căutând un echilibru în dezvoltarea subiectelor tratate, și păstrând regrete pentru cele care nu au putut fi prezentate sau nu au fost dezvoltate suficient din rațiuni impuse de dimensiunea oricărui curs universitar de un semestru. Acest manual încearcă să furnizeze cititorilor atât o sumă de cunoștințe fundamentale, care să îi ajute pe studenți să înțeleagă ce este important în domeniu, cât și o serie de bune practici de care toți avem nevoie.

Această carte este structurată pe 8 capitole.

Primul capitol este dedicat limbajelor de programare și translațoarelor. După ce sunt definite unele noțiuni de bază, în cadrul capitolului sunt prezentate conceptele fundamentale din domeniul limbajelor de programare, istoricul și câteva modalități de clasificare a limbajelor de programare. Capitolul continuă cu prezentarea rolului compilatoarelor, asamblatoarelor, editoarelor de legături, și încărcătoarelor, pentru ca ulterior să fie prezentate unele elemente de implementare a acestora. În cadrul unui subcapitol dedicat compilatoarelor sunt prezentate unele elemente de teoria compilatoarelor și sunt descrise rolul și unele elemente de implementare pentru elementele de bază ale unui compilator: analiza lexicală, analiza sintactică, analiza semantică, generarea codului intermediar, tabela de simboluri, tratarea erorilor în compilare, optimizarea codului, generarea

programului obiect, etc. Capitolul se încheie cu o prezentare a unor clase reprezentative de aplicații software.

Al doilea capitol este dedicat prezentării unor concepte fundamentale din domeniul sistemelor de operare. După unele noțiuni introductive sunt prezentate evoluția sistemelor de calcul și a unor sisteme de operare semnificative. Odată cu prezentarea evoluției sistemelor de operare sunt introduse și explicate și unele dintre conceptele de bază din domeniul sistemelor de operare. Capitolul continuă cu prezentarea unor modalități diferite de clasificare a sistemelor de operare. În subcapitole separate sunt prezentate apoi componentele sistemelor de operare, sarcinile sistemelor de operare, structura sistemelor de operare și elemente de proiectare și implementare a sistemelor de operare.

Ca o exemplificare a conceptelor teoretice prezentate în Capitolul 2, Capitolul 3 este dedicat sistemelor de operare de tip Microsoft Windows, cu focalizare pe Windows 7. Sunt prezentate, în acest context, probleme referitoare la administrarea fișierelor și a folderelor, lucrul cu aplicațiile Windows, administrarea imaginilor și secvențelor video, configurarea calculatorului folosind Panoul de control, administrarea resurselor hardware și software și utilizarea resurselor Internet.

Capitolul 4 tratează problematica, atât de actuală, a securității informatice. În primele subcapitole sunt definite noțiunile de securitate și securitate informatică, sunt prezentate vulnerabilitățile, amenințările și riscurile din domeniul informaticii și apoi sunt descrise câteva dintre tipurile de activități desfășurate de către hackeri, crima organizată, teroriști, sau extremiști ideologici și politici, și de către state care sponsorizează agresiuni cibernetice. Un alt subcapitol este dedicat securității informatice în sistemele de operare Microsoft Windows. Într-un alt subcapitol sunt prezentate probleme de bază referitoare la criptografia computațională. După ce sunt definite noțiunile fundamentale și este făcută o sinteză a istoricului criptografiei și criptanalizei, sunt prezentate elementele de bază legate de criptografia modernă: criptografia cu chei simetrice și criptografia asimetrică. Ultimul subcapitol prezintă modul în care poate fi asigurată securitatea informatică prin intermediul sistemelor de calcul tolerante la defectări. După ce sunt definite noțiunile de evitarea defectării și toleranța la defectări, sunt prezentate domeniile în care toleranța la defectări este o necesitate, și se face o caracterizare generală a metodelor și tehnicilor utilizate pentru implementarea toleranței la defectări. Subcapitolul conține și o sinteză, care arată evoluția în timp a metodelor, tehnicilor, structurilor și arhitecturilor sistemelor de calcul tolerante la defectări. Un subcapitol separat este dedicat detecției și localizării defectelor în sistemele de calcul tolerante la defectări. În continuare sunt abordate conceptele structurale și arhitecturale fundamentale din domeniul sistemelor de calcul tolerante la defectări. În cadrul lui este definită redundanța și este introdus conceptul de comportament cu defectări permise. Sunt caracterizate,

de asemenea, cele două tehnici de bază utilizate pentru mascarea defectelor: mascarea ierarhică și mascarea de tip grup a defectelor. În cadrul subcapitolului sunt dedicate subcapitole separate problemelor referitoare la partiționarea ierarhică și restabilirea funcțională după defectare. Următoarele subcapitole sunt dedicate tehnicilor și metodelor hardware și software de tolerare a defectelor. În aceste subcapitole se definesc noțiunile de unități hardware și software înlocuibile, cu exemplificări pentru sisteme reprezentative, și sunt realizate sinteze cu cele mai reprezentative tehnici și metode hardware și software de tolerare a defectelor

Capitolul 5 încearcă să facă o sinteză a elementelor care fac ca Microsoft Office să poată fi considerat un pachet integrat de aplicații. După o prezentare generală a elementelor și a unor versiuni de pachete integrate Office, sunt exemplificate câteva elemente comune tuturor aplicațiilor Office.

Capitolul 6 prezintă funcționarea și utilizarea editoarelor de documente, cu exemplificare pe Microsoft Office Word 2007. Sunt prezentate funcțiile benzilor (panglicilor) cu comenzi ale programului Word 2007, modurile de vizualizare și navigare în documente, modul în care se lucrează cu documentele și operații ce se pot efectua asupra textului, cum se lucrează cu tabele, elemente de tip WordArt, și cum se tipăresc documente.

În Capitolul 7 sunt prezentate funcțiile de bază ale unor programe folosite pentru prezentări, cu exemplificare pe Microsoft Office PowerPoint 2007. După ce sunt prezentate elementele ferestrei PowerPoint, este arătat modul în care se elaborează o prezentare. Subcapitole distincte sunt dedicate operațiilor efectuate asupra textului unei prezentări și operațiilor cu obiectele grafice. Capitolul se încheie cu explicarea modului în care se pot utiliza șabloanele de proiectare, scheme de culori diferite, diverse scheme de animație și de tranziție, și modul în care se pot vizualiza prezentările realizate.

Ultimul capitol abordează problema calculului tabelar, și a posibilităților oferite de Microsoft Office Excel 2007 în acest domeniu. După ce sunt prezentate elementele și operațiile cu registrele de lucru și foile de lucru, un subcapitol separat este dedicat operațiilor la nivelul elementelor unei foi de lucru. Capitolul continuă cu introducerea, editarea și formatarea informațiilor dintr-o foaie de lucru. În cadrul subcapitolului referitor la formulele din Excel sunt operatorii, operanzii și modul de editare a formulelor. Un subcapitol extins este dedicat funcțiilor din Excel. În cadrul acestuia este prezentat modul de definire a intervalelor și funcțiilor și sunt prezentate funcțiile mai importante din domeniile statistică, matematică și trigonometrie, funcțiile logice, funcțiile de tip text, funcțiile de tip dată și timp și funcțiile de tip financiar. Capitolul continuă cu utilizarea diagramelor și graficelor în Excel. În ultimul subcapitol este arătat modul cum pot fi utilizate informațiile din Excel ca o bază de date, modalitatea de

definire a formularelor în Excel, de sortare și de filtrare a informațiilor din bazele de date Excel.

Autorul este conștient de faptul că orice carte, oricât ar fi ea de bine elaborată, nu poate răspunde tuturor întrebărilor dintr-un domeniu atât de vast. Pentru aceasta toate sugestiile sunt binevenite și vor fi analizate în eventualitatea redactării unei noi ediții îmbunătățite.

1 LIMBAJE DE PROGRAMARE ȘI TRANSLATOARE

1.1 NOȚIUNI INTRODUCATIVE

Dezvoltarea informațională a omenirii poate fi considerată ca având cinci faze de "inovare" a informației. Prima fază corespunde posibilităților "limbajelor vorbite"; a doua fază este legată de apariția literelor și dezvoltarea scrierii; a treia fază este determinată de apariția tiparului și dezvoltarea tipăririi, prevestind prima "explozie informațională și științifică" în interiorul națiunilor și depășind frontierele dintre națiuni. A patra fază constă în nașterea și dezvoltarea telecomunicațiilor, de la primul purtător de cod telegrafic și până în zilele noastre, cu transmisii ale vocii, datelor, textelor și imaginilor. A cincea fază - care a început odată cu calculatoarele electronice - este caracterizată prin posibilitatea de memorare (stocare), prelucrare și transmitere a unei mari cantități de informație complexă, care nu mai poate fi manipulată prin mijloace convenționale. Datorită acestui fapt a crescut în mod remarcabil câmpul de activități intelectuale ale omului și se prevede, pentru viitorul apropiat, apariția unei noi organizări sociale, bazată pe activitatea intelectuală și progresul tehnic, denumită "societatea informațională". În acest nou tip de societate producția de informație va juca un rol mult mai important decât producția industrială, iar calculatoarele și telecomunicațiile vor fi combinate sub formă de "sisteme de prelucrare a informației". În literatura de specialitate este tot mai mult folosită denumirea de "epocă (eră) informatică" pentru a caracteriza această a cincea fază a dezvoltării informaționale a omenirii [ȘERB11b].

Drumul spre producția de informației a fost și este însă unul lung și anevoios. În secolul XVIII oamenii de știință, în dorința lor de a descoperi și de a folosi legile naturii, au ajuns să înțeleagă, în cadrul proceselor fizice, pe lângă natura substanțială a acestora și latura lor energetică. Bazându-se pe progresele din fizică, această nouă abordare a condus la descoperirea unor legi care guvernează schimbările energetice din sisteme, și la identificarea unor invarianți ai acestor schimbări. Ulterior, s-a constatat că punctul de vedere substanță-energie nu este suficient pentru explicarea unor procese naturale. Exemplul tipic îl reprezintă anumite procese din organismele vii, în care schimbările substanțial-energetice, de netăgăduit ca existență, prezintă un aspect departe de a fi reprezentative pentru sistemele respective. Ereditatea, metabolismul, activitatea creierului nu pot fi explicate și studiate ca transformări de energie sau substanță deoarece acestea reprezintă aspecte secundare ale proceselor respective. Mult mai important în asemenea procese este aspectul lor *informațional*. Transmiterea caracterelor părinților prin coduri la nivelul genelor, circulația hormonilor în organism, prelucrarea reprezentărilor la nivelul cortexului sunt în primul rând caracterizabile informațional. În acest fel, conceptul de *informație*, determinant în

înțelegerea multor procese din lumea înconjurătoare, se plasează pe același nivel cu conceptele de substanță și energie constituind o dimensiune aparte ce trebuie asociată lumii materiale. Într-adevăr, informația nu depinde de tipul suportului substanțial-energetic pe care se află depusă, chiar dacă ea este operațională numai pe un asemenea suport. Informația își evidențiază, însă, prezența și își delimitează sensul în corelație cu un alt concept integrator, cel de sistem [ȘERB87].

O interpretare foarte intuitivă a conceptului de sistem ar putea fi următoarea: un sistem este ceva (o structură, de exemplu) în care se introduce la un moment dat ceva (materie, energie, sau informație, de exemplu) și din care rezultă spre exterior, la un moment de timp, ceva (informație, energie, de exemplu).

Această interpretare nu este precisă, totuși permite punerea în evidență a următoarelor elemente:

- sistemul S include o mulțime T de intervale de timp asociate; în fiecare moment $t \in T$ de intervale de timp sistemul S primește un semnal de intrare $x(t)$ și generează un semnal de ieșire $y(t)$;

- considerând că valorile semnalelor de intrare aparțin unei mulțimi date X , înseamnă că la fiecare moment $t \in T$ valoarea $x(t)$ poate fi aleasă din mulțimea X ;

- funcțiile de forma $u: [t_1, t_2) \rightarrow X$, unde $[t_1, t_2) \subseteq T$ se numesc “segmente de intrare” și reprezintă succesiuni (secvențe) în timp ale semnalelor de intrare; mulțimea segmentelor de intrare va fi notată U ;

- fiecare valoare a semnalelor de ieșire $y(t)$ a sistemului S aparține unei mulțimi Y și în general pot fi impuse anumite restricții semnalelor de ieșire, de forma $v: [t_2, t_3) \rightarrow Y$; mulțimea semnalelor de ieșire va fi notată V și $v \in V$ reprezintă o succesiune (secvență) de ieșire $y(t)$ pentru $t \in [t_2, t_3) \subseteq T$.

Pe baza celor afirmate mai sus rezultă că un sistem poate fi descris printr-un ansamblu de attribute măsurabile $x(t)$ și $y(t)$, numite variabile accesibile, care reprezintă singurele legături ale sistemului cu exteriorul.

Se definește noțiunea de **obiect abstract** ca fiind tripletul:

$$A = (X, Y, g) \tag{1.1.}$$

unde $g: X \rightarrow Y$ este o funcție care asociază la fiecare moment de timp cele două mărimi $x(t)$ și $y(t)$ printr-o relație $y = g(x)$.

Definiția dată obiectului abstract este o exprimare formală a faptului că orice interacțiune cu un “obiect” (material sau imaterial) implică varierea unora dintre attributele acestui obiect și observarea variațiilor care rezultă la celelalte attribute. Attributele care sunt variate $x(t)$ joacă rolul de intrări (cauze), iar variațiile rezultate $y(t)$ sunt ieșiri (efecte).

Un sistem S poate fi modelat printr-un obiect abstract dacă analiza acestuia este situată la **nivel global**. Pentru modelarea cât mai completă a sistemului cel mai adesea este necesară cunoașterea structurii sistemului și

realizarea analizei la **nivel structural**. În continuare vor fi introduse elementele necesare analizei structurale a sistemelor.

În general, mărimea de ieșire $y(t)$ a sistemului **S** depinde de “**starea**” sistemului **S**, înțelegând prin “**stare prezentă**” (“**stare actuală**”) în mod intuitiv acea parte a evoluției trecute și prezente a lui **S** care este semnificativă pentru determinarea mărimii de ieșire.

De asemenea, mărimea de ieșire $y(t)$ depinde și de mărimea de intrare $x(t)$, în cele mai multe cazuri. Notând $z(t)$ “starea actuală” și înțelegând prin aceasta “ultima” stare a evoluției, rezultă o mulțime **Z** de “stări” ale sistemului **S**; cunoașterea stării actuale $z(t)$ trebuie să fie posibilă dacă se dispune de o “**stare inițială**” $z(t_0)$ și de segmentul de intrare $u = u[t_0, t)$, ori de câte ori $t_0 \leq t$; această cerință înseamnă că s-a introdus o direcționare preferențială a timpului, astfel încât trecutul precede viitorul.

Sistemul **S**, înțeles în sensul arătat mai sus, va fi numit “**sistem dinamic**” (sau “**neanticipativ**”); astfel noțiunea matematică de “sistem dinamic” este destinată descrierii fenomenelor “cauzale”, din trecut către viitor (trecutul influențează viitorul) [ȘERB96].

Definițiile date mai sus sistemelor nu sunt cu nimic restrictive. Pe baza lor putem vorbi și despre sisteme naturale, și despre sisteme artificiale (create de om), și despre sisteme mixte. În general, orice sistem interacționează cu mediul pe trei paliere: substanțial, energetic și informațional. După cum se știe, la un moment dat, în funcție de interesul observatorului, unele paliere pot fi estompate ca importanță sau chiar ignorate, iar altele subliniate. În cele ce urmează ceea ce interesează este palierul informațional. Un sistem real „văzut” numai prin intermediul schimburilor de informații devine un *sistem informațional*. Sistemul informațional este astfel o imagine constuită astfel încât să reflecte în mod esențial schimbul de informații în cadrul sistemului real și între acesta și mediul înconjurător. Adesea un sistem informațional se prezintă ca o structură ierarhică a componentelor: fiecare din acestea este un sistem de un nivel mai coborât care are componente ce pot fi și ele sisteme etc. Componentele ultimului nivel sunt indivizibile din punct de vedere al criteriului care a reunit componentele în sistem. Rezultă că o componentă poate fi definită, din punct de vedere al sistemului din care face parte, exclusiv prin interfețele ei, independent de structura restului sistemului. Alcătuirea internă a componentei trebuie să justifice comportarea ei externă, comportare la nivelul interfețelor. Abordarea sistemică se dovedește extrem de fructuoasă atât în cercetarea lumii înconjurătoare, cât și în realizarea de produse noi. Un asemenea produs poate fi văzut ca un sistem în care fiecare componentă legată slab de celelalte poate fi proiectată și realizată separat plecând de la interfețe fixe. Pentru proiectarea componentei este util să se considere că ea însăși este un sistem alcătuit din componente ce pot fi proiectate separat etc. Abordarea sistemică folosește astfel la depășirea obstacolului creat de

complexitatea unor sisteme. Un caz extrem de interesant este cel al produselor de programare văzute ca sisteme complexe. În prezent, abordarea sistemică în programare stă la baza proiectării prin detalieri succesive și a programării structurate.

Privind din perspectiva mașinilor de calcul organizate sistemic, pe mai multe nivele, cel mai simplu mod de a privi un sistem de calcul este acela în care sistemul de calcul poate fi văzut ca o mașină cu trei nivele. Pe nivelul cel mai de jos este hardware-ul, iar pe cel mai de sus este software-ul, nivelul intermediar fiind constituit din setul de instrucțiuni al calculatorului (Fig. 1.1).



Fig. 1.1 Calculator cu trei nivele

Analizând calculatoarele, în modul cel mai general de abstractizare, nivelul de bază a oricărui calculator este, așadar, **hardware**-ul calculatorului. Hardware-ul calculatoarelor actuale este format din circuite electronice, microprocesoare, memorii, dispozitive de intrare/ieșire, etc., deci din obiecte fizice, tangibile, de tipul circuitelor integrate, memoriilor, plăcilor cablate, surselor de alimentare, monitoarelor, imprimantelor, etc. Hardware-ul, ar reprezenta deci ansamblul componentelor fizice și tehnice cu ajutorul cărora datele și programele se pot introduce, stoca, prelucra, livra, verifica, transmite, etc. Hardware este, deci, termenul general care desemnează resursele fizice (circuitele, dispozitivele și echipamentele componente) ale unui calculator numeric.

Nivelul superior este constituit din **software**, care constă din programe scrise pe baza unor algoritmi. Software reprezintă ansamblul componentelor logice de tipul programelor, procedurilor, rutinelor, modulelor, etc., care dirijează funcționarea componentelor hard și asigură prelucrarea datelor, cu scopul de a rezolva diverse probleme concrete. Există două tipuri de software: software de sistem și software de aplicații. Cea mai importantă componentă a software-ului de bază este **sistemul de operare**, cel care administrează și controlează hardware-ul și furnizează mediul de operare pentru toate programele. **Software-ul de aplicații (software-ul de asistență pentru dezvoltarea aplicațiilor)** este numele dat tuturor programelor care asistă utilizatorii pentru a-și realiza prelucrărilor dorite.

Nivelul care face posibil dialogul hardware-ului cu software-ul este **setul de instrucțiuni** (limbajul mașină) pe care este capabil să-l execute un calculator.