

# TEMATICĂ DE EXAMEN

---

**Verificarea și Validarea Sistemelor Soft**  
**[12 Mai 2020]**

Lector dr. Camelia Chisăliță-Crețu  
Universitatea Babeș-Bolyai Cluj-Napoca

# Conținut

- Curs 01. Verificare și Validare. Inspectare
- Curs 02. Testare. Testare Black-box
- Curs 03. Testare White-Box
- Curs 04. Niveluri de testare
- Curs 05. Execuție simbolică
- Curs 06. Automation Testing + Performance Testing (Evozon)
- Curs 07. Corectitudine
- Curs 08. Raportarea bug-urilor
- Curs 09. Verificarea modelelor
- Curs 10. QA. QC (Endava)

# Curs 01. Verificare și Validare. Inspectare

- **Concepte, caracteristici, asemănări și diferențe:**
  - verificare, validare; verificare vs. validare;
  - eroare, defect/bug, defecțiune; eroare vs. defect/bug vs. defecțiune;
  - stakeholders, calitate, QA, QC;
  - Analiza statică vs analiza dinamică;
  - HbT, motivație și aplicare HbT;
  - inspectare, walkthroughs, technical review:
    - descriere; echipa de membri: enumerare și descrierea rolurilor, activități asociate: enumerare; avantaje;
  - informal review:
    - descriere; echipa de membri: enumerare și descrierea rolurilor, activități asociate: enumerare; avantaje;
  - pair-programming:
    - caracteristici, avantaje.

# Curs 02. Testare. Testare Black-Box

- **Testare:**
  - false definiții ale testării (3); definiții ale testării (4);
  - terminologie: program, program testat, caz de testare;
  - tipuri de testare: exhaustivă, selectivă;
  - reguli de raportare a unui bug;
  - ciclul de viață al unui bug (diagrame, descriere);
- **Testare black-box:**
  - definiție, caracteristici;
  - ECP, BVA, ECP vs. BVA, ECP + BVA;
  - aplicarea ECP și BVA pentru probleme concrete;
  - avantaje și dezavantaje.

# Curs 03. Testare White-Box

- **testare white-box:**
  - definiție, caracteristici, avantaje și dezavantaje;
  - CFG (definiție și construire), drumuri independente (definiție), CC (definiție, 3 moduri de calcul);
  - construirea CFG, determinarea drumurilor independente și calculul CC (3 moduri) pentru metode concrete;
  - criteriile de acoperire **apc, sc, dc, cc, dcc, mcc** și **lc** (definiție, compararea a două criterii, relațiile existente între criterii);
  - testare black-box vs. testare white-box.

# Curs 04. Niveluri de testare

- **Niveluri de testare. Definiții și caracteristici:**
  - testare unitară;
  - testare de integrare;
    - 4 tehnici (big-bang, top-down, bottom-up, sandwich), descriere, comparare;
  - testare de sistem;
    - testare funcțională;
    - 5 tipuri de testare non-funcțională (volume, stress, load, usability, security) [Mye04].
  - testare de acceptare;
    - alpha testing, beta testing.
- **Tip de testare vs Nivel de testare. Definiție și caracteristici:**
  - re-testare;
  - testare de regresie.

# Curs 05. Execuție simbolică

- **Execuție simbolică:**

- definiții: valoare simbolică, execuție simbolică, stare simbolică (variabile + pc + counter);
- descriere: comutativitate CE – SE;
- execuția simbolică a structurilor secvențiale, alternative și repetitive;
- SET: definiție, proprietăți, construire SET, CFG vs. SET;
- execuția simbolică și generarea cazurilor de testare.

# Curs 06. Test Automation. Performance Testing

- **Automation testing:**
  - What is test automation?
  - Why to automate?
  - What to automate?
  - Why do we still need manual testing?
- **Performance testing:**
  - *inclusă în tematica cursului 04.*





# Curs 07. Corectitudine

- **Corectitudine I. Metoda lui Floyd (metoda aserțiunilor inductive)**
  - elementele necesare construirii condițiilor de verificare și condițiilor de terminare;
    - puncte de tăietură, drumuri ( $d_{ij}$ ), predicate invariante, condiții de parcurgere a drumurilor, funcții de transformare a variabilelor pe drumurile parcurse, funcțiile  $u_i(X, Y)$  asociate punctelor de tăietură;
  - demonstrarea parțial corectitudinii, terminării și total corectitudinii ([Fre10], Cap.1) – **2 probleme**:
    - **determinarea celui mai mare divizor comun a două numere naturale** (Seminar 5);
    - **căutarea unei valori într-un șir ordonat** (Seminar 5);
- **Corectitudine II. Teoria lui E. Dijkstra**
  - rafinare: definirea regulilor;
  - rafinare algoritmi din specificații (**4 2 probleme, Seminar 06**):
    - **împărțire întreagă (cât și rest)**;
    - **rădăcină pătrată**;
    - ~~• înmulțire prin adunări repetate;~~
    - ~~• cel mai mare divizor comun a două numere naturale.~~

# Curs 08. Raportarea bug-urilor

- **Ciclul de viață al unui bug:**
  - Cele două variante discutate la curs (simplu și detaliat);
- **RIMGEA**
  - descrierea semnificației acronimului:
    - replicate,
    - isolate,
    - maximize,
    - generalize,
    - externalize,
    - communicate it clear;
  - Tipuri de bug-uri:
    - Coding bug;
    - Design bug;
    - Coding bug vs design bug.

# Curs 09. Verificarea modelelor

- **Verificarea modelelor**

- verificare formală: definiție, etape de realizare, avantaje, dezavantaje;
- sistem de tranziții: definiție;
- definiția unor proprietăți ale sistemului: correctness, liveness, safety, real-time, deadlock;
- logica temporală (TL): definiție, operatori;
- logica temporală liniară (LTL): definiție, operatori, specificarea proprietăților (**exemplele de la curs, folosind operatori NuSMV**);
- logica temporală computațională (CTL): definiție, operatori, specificarea proprietăților (**exemplele de la curs, folosind operatori NuSMV**).

Logica	Operator	Semnificație	NuSMV
propozițiilor	$\neg \wedge \vee \rightarrow \leftrightarrow$	Not, And, Or, Imply, Equivalence	!, &,  , ->, <->
TL	$\square$	Always/ <b>G</b> lobally	<b>G</b>
TL	$\diamond$	Eventually/ <b>F</b> inally/ <b>F</b> uture	<b>F</b>
LTL	$\circ$	Next/ <b>NeX</b> t State	<b>X</b>
LTL	<b>U</b>	Until/ <b>U</b> ntil	<b>U</b>
CTL	$\exists$	Exists/ <b>E</b> xists	<b>E</b>
CTL	<b>V</b>	For All/ For <b>A</b> ll	<b>A</b>

# Curs 10. QA. QC

- **NU este inclus în tematica de examen!**



# Curs 11. Curs 12

- **Curs 11:**
  - **Recapitulare pentru examen + simulare examen scris.**
- **Curs 12:**
  - **Prezentare referate/tutoriale.**