

FI A DISCIPLINEI

1. Date despre program

| | |
|---------------------------------------|--|
| 1.1 Instituția de învățământ superior | Universitatea Babeș-Bolyai |
| 1.2 Facultatea | Facultatea de Matematica și Informatică |
| 1.3 Departamentul | Departamentul de Informatică |
| 1.4 Domeniul de studii | Știința calculatoarelor |
| 1.5 Ciclul de studii | Licență |
| 1.6 Programul de studiu / Calificarea | Ingineria informației |

2. Date despre disciplină

| | | | | | | | |
|--|---|---------------|----------|------------------------|----------|-------------------------|--------------------|
| 2.1 Denumirea disciplinei | Verificarea și Validarea Sistemelor soft | | | | | | |
| 2.2 Titularul activităților de curs | Lect. Dr. Andreea Vescan | | | | | | |
| 2.3 Titularul activităților de seminar | Lect. Dr. Andreea Vescan | | | | | | |
| 2.4 Anul de studiu | 3 | 2.5 Semestrul | 6 | 2.6. Tipul de evaluare | E | 2.7 Regimul disciplinei | obligatoriu |

3. Timpul total estimat (ore pe semestru al activităților didactice)

| | | | | | |
|--|----|--------------------|----|-----------------------|-----|
| 3.1 Număr de ore pe săptămână | 4 | Din care: 3.2 curs | 2 | 3.3 seminar/laborator | 2 |
| 3.4 Total ore din planul de învățământ | 48 | Din care: 3.5 curs | 24 | 3.6 seminar/laborator | 24 |
| Distribuția fondului de timp: | | | | | ore |
| Studiul după manual, suport de curs, bibliografie și notițe | | | | | 24 |
| Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren | | | | | 24 |
| Pregătirea seminarilor/laboratoare, teme, referate, portofolii și eseuri | | | | | 24 |
| Tutoriat | | | | | 6 |
| Examinări | | | | | 24 |
| Alte activități: | | | | | 0 |
| 3.7 Total ore studiu individual | | 102 | | | |
| 3.8 Total ore pe semestru | | 150 | | | |
| 3.9 Numărul de credite | | 5 | | | |

4. Precondiții (acolo unde este cazul)

| | |
|-------------------|---|
| 4.1 de curriculum | • |
| 4.2 de competențe | • |

5. Condiții (acolo unde este cazul)

| | |
|--|---|
| 5.1 De desfășurare a cursului | • |
| 5.2 De desfășurare a seminarului/laboratorului | • |

6. Competențele specifice acumulate

| | |
|--------------------------------|--|
| Competențe profesionale | <ul style="list-style-type: none"> • Cunoașterea, înțelegerea și utilizarea conceptelor teoretice de bază din informatică; • Capacitatea de a lucra independent pentru rezolvarea unor probleme. |
| Competențe transversale | <ul style="list-style-type: none"> • Abilități de programare îmbunătățite: depanarea și corectarea erorilor de compilare. |

7. Obiectivele disciplinei (reie îndin grila competențelor acumulate)

| | |
|---------------------------------------|--|
| 7.1 Obiectivul general al disciplinei | <ul style="list-style-type: none"> • Înțelegerea noțiunilor de algoritm parțial și total corect; • Formarea deprinderilor de proiectare a algoritmilor în paralel cu demonstrarea corectitudinii lor; • Cunoașterea metodelor de testare și verificare a sistemelor soft; • Formarea deprinderilor de proiectare a programelor corecte din specificații; • Formarea unui stil modern de programare. |
| 7.2 Obiectivele specifice | <ul style="list-style-type: none"> • Studentii vor ști cum se desfășoară și care sunt pașii unei inspecții, fie a codului sursă fie a specificației din fiecare etapă de dezvoltare a sistemului soft. • Studentii vor ști să prevadă încă din faza de specificare și proiectare crearea unor cazuri de testare care să-i ajute la dezvoltarea unui sistem soft mai robust. |

8. Conținuturi

| 8.1 Curs | Metode de predare | Observații |
|---|--|------------|
| 1. Verificarea și validarea. Concepte; | Prezentare, Demonstrații, Problematizare | |
| 2. Testarea programelor (1): conceptul de testarea programelor; testarea unitară, criteriile de testare. testarea blackbox și whitebox; | Prezentare, Demonstrații, Problematizare | |
| 3. Testarea programelor (2): tipuri de testare (T de integrare, T. de sistem, T. de regresie, T de acceptanță), automatizarea testării. | Prezentare, Demonstrații, Problematizare | |
| 4. Inspectarea programelor | Prezentare, Demonstrații, | |

| | | |
|---|---|------------|
| | Problematizare | |
| 5. Executia simbolica | Prezentare, Demonstratii, Problematizare | |
| 6. Model checking | Prezentare, Demonstratii, Problematizare | |
| 7. Teoria corectitudinii programelor. Evolutia conceptului de corectitudine. Contributiilor lui Floyd, Hoare, Dijkstra, Gries, Droomey, Morgan | Prezentare, Demonstratii, Problematizare | |
| 8. Specificarea programelor. Metoda lui Floyd de demonstrare a corectitudinii. Dijkstra: cea mai slaba preconditionie. Rafinare in pasi succesivi din specificare. Metoda axiomatica a lui Hoare. | Prezentare, Demonstratii, Problematizare | |
| 9. Compararea metodelor de verificare (corectitudine-inspectare-testare-executie simbolica) Verificare si validare: Cum? De ce? Cand? | Prezentare, Demonstratii, Problematizare | |
| 10. Cleanroom. Calitatea Programelor. | Prezentare, Demonstratii, Problematizare | |
| 11. Calitate, SPI, SQA,CMM. | Prezentare, Demonstratii, Problematizare | |
| 12. Consecintele teoriei corectitudinii programarii asupra programarii. Stil in programare. | Prezentare, Demonstratii, Problematizare | |
| | | |
| | | |
| Bibliografie | | |
| 1. BALANESCU T., Corectitudinea programelor, Editura tehnica, Bucuresti 1995. | | |
| 2. DIJKSTRA, E., A constructive approach to the problem of program correctness, BIT, 8(1968), pg.174-186. | | |
| 3. DIJKSTRA, E., Guarded commands, nondeterminacy and formal derivation of programs, CACM, 18(1975), 8, pg.453-457. | | |
| 4. DROMEY G., Program Derivation. The Development of Programs From Specifications, Addison Wesley Publishing Company, 1989. | | |
| 5. FRENTIU, M., Verificarea corectitudinii programelor, Ed.Univ."Petru-Maior", 2001. | | |
| 6. GRIES, D., The Science of Programming, Springer-Verlag, Berlin, 1981. | | |
| 7. HOARE, C.A.R., An axiomatic basis for computer programming, CACM, 12(1969), pg.576-580, 583. | | |
| 8. Morgan, C., Programing from Specifications, Prentice Hall, NewYork, 1990. | | |
| B. Internet | | |
| 8.2 Seminar / laborator | Metode de predare | Observa ii |
| 1. S1: Specificatii si Inspectare L1: Analiza statica folosind ESCJava2, JML | Prezentare, conversatie, Problematizare, Descoperire, Studiu individual, Exercitii | |
| S2: Cazuri de testare utilizand metodele de testare Black-box Testing (BBT) si White-box (WBT) L2:Black-box Testing | Prezentare, conversatie, Problematizare, Descoperire, Studiu individual, Exercitii | |

| | | |
|--|--|--|
| S3: Lucrare de control 1 + BBT si WBT L3:White-Box Testing | Prezentare, conversatie, Problematizare, Descoperire, Studiu individual, Exercitii | |
| S4: Model checking+ Corectitudine L4:Model cheking | Prezentare, conversatie, Problematizare, Descoperire, Studiu individual, Exercitii | |
| S5: Lucrare de control 2 + Corectitudine L5:Testare GUI, Web app. | Prezentare, conversatie, Problematizare, Descoperire, Studiu individual, Exercitii | |
| S6:Inspectare L6:Inspectare | Prezentare, conversatie, Problematizare, Descoperire, Studiu individual, Exercitii | |
| | | |
| | | |
| Bibliografie | | |

9. Coroborarea con inuturilor disciplinei cu a tept rile reprezentan ilor comunit ii epistemice, asocia iilor profesionale i angajatori reprezentativi din domeniul aferent programului

- Studentii vor sti cum sa aplice metodele de testare pentru un produs soft, metode de testare utilizare in industrie.
- Studentii vor invata diverse metode de verificare si validare ale unui sistem soft.

10. Evaluare

| Tip activitate | 10.1 Criterii de evaluare | 10.2 metode de evaluare | 10.3 Pondere din nota final |
|--|---|--|-----------------------------|
| 10.4 Curs | La sfarsitul semestrului se va da o nota E pe examinarea scrisa. | Examen scris | 50 |
| 10.5 Seminar/laborator | Activitarea de seminar, constand in participarea la discutii si la rezolvarea unor exercitii, va fi notata cu o nota S. | Lucrare de control 1+ Lucrare de control 2+ Activitatea de Seminar | 25 |
| | Activitatea de laborator va fi notata cu nota L. | | 25 |
| 10.6 Standard minim de performan | | | |
| <ul style="list-style-type: none"> • Studentii vor invata si aplica metdele de testare ale unui produs soft. • Studentii vor aplica diverse metode de verificare (testare, inspectare, model checking) pentru stabilirea corectitudinii unui algoritm. • Cel putin nota 5 (notare de la 1 la 10) la fiecare componenta din nota finala (examenul scris, | | | |

laborator, activitatea de seminar).

Data completării

30.04.2013

Titular de curs

Lect. Dr. Andreea Vescan

Titular de seminar

Lect. Dr. Andreea Vescan

Data avizării în departament

.....

Director de departament

Prof. Dr. Bazil Parv