



Universitatea Națională de Știință și Tehnologie Politehnica București
Facultatea de Electronică, Telecomunicații și
Tehnologia Informației



FIȘA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Națională de Știință și Tehnologie Politehnica București
1.2 Facultatea	Electronică, Telecomunicații și Tehnologia Informației
1.3 Departamentul	Electronică Aplicată și Ingineria Informației
1.4 Domeniul de studii	Calculatoare și Tehnologia Informației
1.5 Ciclu de studii	Masterat
1.6 Specializarea	Sisteme Inteligente și Vedere Artificială

2. Date despre disciplină

2.1 Denumirea disciplinei (ro) (en)	Arhitectura informației						
2.2 Titularul activităților de curs	S.I./Lect. Dr. George Valentin STOICA						
2.3 Titularul activităților de seminar / laborator	S.I./Lect. Dr. George Valentin STOICA						
2.4 Anul de studiu	1	2.5 Semestrul	II	2.6. Tipul de evaluare	E	2.7 Regimul disciplinei	Ob
2.8 Tipul disciplinei	F	2.9 Codul disciplinei	4	2.10 Tipul de notare	Nota		

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

3.1 Număr de ore pe săptămână	2.5	Din care: 3.2 curs	1.50	3.3 seminar/laborator	1
3.4 Total ore din planul de învățământ	35.00	Din care: 3.5 curs	21	3.6 seminar/laborator	14
Distribuția fondului de timp:					ore
Studiul după manual, suport de curs, bibliografie și notițe Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate Pregătire seminarii/ laboratoare/proiecte, teme, referate, portofolii și eseuri					34
Tutorat					0
Examinări					6
Alte activități (dacă există):					0
3.7 Total ore studiu individual	40.00				
3.8 Total ore pe semestru	75				
3.9 Numărul de credite	3				

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	Nu este cazul
4.2 de rezultate ale învățării	Acumulare cunoștințe de Structuri de date si algoritmi Programare obiect-orientata Sisteme de operare Calcul paralel

5. Condiții necesare pentru desfășurarea optimă a activităților didactice (acolo unde este cazul)



5.1 Curs	Cursul se va desfășura într-o sală dotată cu videoproiector.
5.2 Seminar/ Laborator/Proiect	Laboratorul se va desfășura într-o sală cu dotare specifică, ce trebuie să includă: pc-uri, mediu de dezvoltare software precum MS Visual Studio, GPU precum nVidia GTX

6. Obiectiv general *(Se referă la intențiile profesorilor pentru studenți, la ceea ce studenții vor fi învățați în timpul cursului. Oferă o orientare cu privire la locul cursului în cadrul domeniului științific abordat, precum și la rolul pe care acesta îl are în cadrul specializării studiate. Vor fi descrise de o manieră generală tematicile abordate, justificarea includerii cursului în planul de învățământ al specializării studiate etc.)*

Cursul are ca obiectiv aprofundarea de către studenți a conceptelor cu privire la limbajele, structurile și arhitecturile de calcul de performanță utilizate în domeniul tehnologiei informației. Cursul prezintă sintetic limbajele utilizate în platformele de prelucrare a informației precum C/C++/Java/Python/.Net și continuă cu arhitecturile de calcul paralele și distribuite cloud și GPU. Partea a doua a cursului aprofundează arhitectura GPU și mai concret arhitectura CUDA ca și platforma de calcul de înaltă performanță.

Aplicațiile de laborator au ca obiectiv evidențierea practică a conceptelor și noțiunilor specifice calculului și procesării informației. Folosind drept modalități de experimentare dezvoltarea de aplicații CUDA, fie folosind tehnologii precum OpenCL, studenții își pot însuși și pot experimenta cunoștințele dobândite în timpul cursului. Întreg setul de noțiuni dobândite în timpul cursului pot fi experimentate în timpul aplicațiilor de laborator

7. Competențe *(Capacitatea dovedită de a utiliza cunoștințe, aptitudini și abilități personale, sociale și/sau metodologice în situații de muncă sau de studiu și pentru dezvoltarea profesională și personală. Reflecția cerințele angajatorilor.)*

Specifice	Dupa absolvirea acestui curs studentii isi insusesc notiunile de baza si avansate din domeniul programarii utilizand arhitecturi GPU/CUDA. Aceste cunostinte combinate cu parcurgerea aplicatiilor de laborator confera posibilitatea de a dezvolta aplicatii de prelucrare a informatiei. De asemenea, studentii vor fi capabili sa dezvolte tehnici si metode de programare, optimizare si eficientizare a algoritmilor de procesare a informatiei.
Transversale (generale)	Îndeplinirea sarcinilor profesionale cu identificarea exactă a obiectivelor de realizat, a unor factori potențiali de risc, a resurselor disponibile, a aspectelor economico-financiare, condițiilor de finalizare a acestora, etapelor de lucru, timpului de lucru și termenelor de realizare aferente;

8. Rezultatele învățării *(Sunt enunțuri sintetice referitoare la ceea ce un student va fi capabil să facă sau să demonstreze la finalizarea unui curs. Rezultatele învățării reflectă realizările studentului și mai puțin intențiile profesorului. Rezultatele învățării informează studenții despre ceea ce se așteaptă de la ei din punct de vedere al performanței, pentru a obține notele și creditele dorite. Sunt definite în termeni concreți, folosind verbe similare exemplelor de mai jos și indică ceea ce se va urmări prin evaluare. Rezultatele învățării vor fi astfel redactate încât să fie evidențiată clar relația față de competențele definite la punctul 7.)*

Cunoștințe	Rezultatul asimilării de informații prin învățare. Cunoștințele reprezintă ansamblul de fapte, principii, teorii și practici legate de un anumit domeniu de muncă sau de studiu. Pot fi teoretice și/sau faptice. Dupa absolvirea acestui curs studentii isi insusesc notiunile de baza si avansate din domeniul programarii utilizand arhitecturi GPU/CUDA. Aceste cunostinte combinate cu parcurgerea aplicatiilor de laborator confera posibilitatea de a dezvolta aplicatii de prelucrare a informatiei. De asemenea, studentii vor fi capabili sa dezvolte tehnici si metode de programare, optimizare si eficientizare a algoritmilor de procesare a informatiei.
-------------------	---



Aptitudini	<i>Capacitatea de a aplica cunoștințe și de a utiliza know-how pentru a duce la îndeplinire sarcini și a rezolva probleme. Aptitudinile sunt descrise ca fiind cognitive (implicând utilizarea gândirii logice, intuitive și creative) sau practice (implicând dexteritate manuală și utilizarea de metode, materiale, unelte și instrumente).</i> Selectează și grupează informații relevante într-un context dat.
Responsabilitate și autonomie	<i>Capacitatea cursantului de a aplica în mod autonom și responsabil cunoștințele și aptitudinile sale. Demonstrează receptivitate pentru contexte noi de învățare.</i>

9. Metode de predare (Se vor avea în vedere metode care să asigure predarea centrată pe student. Se va descrie modul în care se asigură participarea studenților la stabilirea propriului parcurs de învățare, cum se identifică eventualele rămânări în urmă și ce măsuri remediale se adoptă în astfel de cazuri.)

În cadrul acestei materii, considerând atât orele de curs cât și cele de aplicații, se vor folosi metode de predare atât expositive (prelegerea, expunerea), cât și conversative-interactive, bazate pe modele de învățare prin descoperire facilitate de explorarea directă și indirectă a realității (experimentul, demonstrația, modelarea), dar și pe metode bazate pe acțiune, precum exercițiul, activitățile practice și rezolvarea de probleme.

În activitatea de predare vor fi utilizate prelegeri, în baza unor prezentări Power Point. Prezentările utilizează imagini și scheme, astfel încât informațiile prezentate să fie ușor de înțeles și asimilat.

Prezentările pot fi realizate și cu ajutorul canalelor de comunicații online, pe platforma Teams.

Se va avea în vedere exersarea abilităților de ascultare activă și de comunicare asertivă, precum și a mecanismelor de construcție a feedback-ului. Se va exersa abilitatea de lucru în echipă pentru rezolvarea diferitelor sarcini de învățare.

10. Conținuturi

CURS		
Capitolul	Conținutul	Nr. ore
1	Limbaje de programare 1.1. Prezentarea limbajelor de programare 1.2. Particularități, avantaje, platforme/biblioteci de dezvoltare	2
2	Limbaje de programare 2.1. Particularități, avantaje, platforme/biblioteci de dezvoltare 2.2. C/C++, Java, #c, Python	2
3	Arhitecturi pentru procesarea informației 3.1. Prezentarea arhitecturilor de calcul 3.2. Multiprocesoare, cloud, GPU	2
4	Arhitecturi GPU 4.1. Arhitecturi GPGPU, GPU 4.2. Arhitectura CUDA	2



Universitatea Națională de Știință și Tehnologie Politehnică București
Facultatea de Electronică, Telecomunicații și
Tehnologia Informației



5	Arhitectura CUDA 5.1. Istoric, evoluție 5.2. Descriere 5.3. Concepte specifice CUDA 5.4. Modele de programare CUDA 5.5. Patternuri de programare. Paralelism la nivel de date. Tipuri de memorie CUDA. 5.6. Aplicații. Analiza și optimizare. Performanță.	5
6	Biblioteci bazate pe CUDA 6.1. Prezentarea bibliotecilor bazate pe arhitectura CUDA 6.2. cuBLAS, OpenACC, OpenCL, DNN	4
7	Aplicații 7.1. Aplicații ale prelucrării informației cu implementări GPU/CUDA: prelucrarea imaginilor, inteligența artificială, computer vision, grafică 3D	4
Total:		21

Bibliografie:

David B. Kirk, Wen-mei W. Hwu, Programming Massively Parallel Processors 2nd edition - A Hands-on Approach, Morgan Kaufmann 2013
nVidia CUDA: www.developer.nvidia.com
Matt Pharr, Randima Fernando : GPU Gems 2: Programming Techniques for High-Performance Graphics and General-Purpose Computation, Pearson Education 2005,
http://download.nvidia.com/developer/GPU_Gems_2/CD/Index.html
Jason Sanders, Edward Kandrot, CUDA by Example: An Introduction to General-Purpose GPU Programming, Addison Wesley 2010,
http://www.mat.unimi.it/users/sansotte/cuda/CUDA_by_Example.pdf

LABORATOR

Nr. crt.	Conținutul	Nr. ore
1	Biblioteci GPU/CUDA.	2
2	Dezvoltarea unei aplicații CUDA	2
3	Dezvoltarea unei aplicații CUDA	2
4	Analiza/profilul aplicațiilor CUDA	2
5	Optimizare	2
6	Aplicații: prelucrarea imaginilor	2
7	Aplicații: prelucrarea imaginilor	2
Total:		14

Bibliografie:

CUDA C Programming Guide, <http://docs.nvidia.com/cuda/cuda-c-programming-guide/index.html>
CUDA C Best Practices Guide, <http://docs.nvidia.com/cuda/cuda-c-best-practices-guide/index.html>
David B. Kirk, Wen-mei W. Hwu, Programming Massively Parallel Processors 2nd edition - A Hands-on Approach, Morgan Kaufmann 2013
Jason Sanders, Edward Kandrot, CUDA by Example: An Introduction to General-Purpose GPU Programming, Addison Wesley 2010,
http://www.mat.unimi.it/users/sansotte/cuda/CUDA_by_Example.pdf

11. Evaluare



Tip activitate	11.1 Criterii de evaluare	11.2 Metode de evaluare	11.3 Pondere din nota finală
11.4 Curs	- cunoașterea noțiunilor teoretice fundamentale; - cunoașterea tehnicilor de programare și a arhitecturilor; - cunoașterea platformei GPU/CUDA; - cunoașterea unei biblioteci bazată pe GPU/CUDA	Teste orale, susținute la date fixate în sesiune; subiectele acoperă întreaga materie, realizând o sinteză între parcurgerea teoretică comparativă a materiei și explicitarea prin exerciții a tehnicilor specifice programării platformelor GPU.	50%
11.5 Seminar/laborator/proiect	- cunoașterea modului de proiectare a unui algoritm de analiză a imaginilor pentru rezolvarea unei probleme date; - cunoașterea modului de transpunere în cod a unui algoritm; - demonstrarea funcționării unui algoritm implementat.	Proiectul final de laborator, cuprinzând o componentă teoretică și o componentă practică. Componenta teoretică este verificată prin elaborarea unui referat; componenta practică este evaluată prin verificarea modului de rezolvare (implementare, testare, funcționare) de către student a unor probleme practice.	50%
11.6 Condiții de promovare			
Proiectarea, implementarea, și demonstrarea funcționării unor soluții pentru procesarea imaginilor, computer vision, inteligența artificială folosind biblioteci cu accelerare GPU/CUDA. Obținerea a 50% din punctajul total. Obținerea a 50% din punctajul aferent activității de laborator pe parcursul semestrului.			

12. Coroborarea conținutului disciplinei cu așteptările reprezentanților angajatorilor și asociațiilor profesionale reprezentative din domeniul aferent programului, precum și cu stadiul actual al cunoașterii în domeniul științific abordat și practicile în instituții de învățământ superior din Spațiul European al Învățământului Superior (SEİS)

Procesarea informației necesită cunoașterea arhitecturilor disponibile în prezent pe piață, a limbajelor de programare, a structurilor de date necesare precum și a bibliotecilor existente adaptate diverselor domenii precum prelucrarea imaginilor, computer vision, inteligența artificială, grafica 3D, calcule matematice, algebra. Înțelegerea acestor aspecte împreună cu particularitățile fiecărei arhitecturi hardware creează premisele unor implementări eficiente, optimizate și adaptate cerințelor de calcul din ce în ce mai crescute odată cu creșterea complexității aplicațiilor. Prelucrarea rapidă a informației necesită utilizarea unor platforme adaptate, fie ca se utilizează arhitecturi cloud, arhitecturi GPU sau platforme mobile. Procesarea informației necesită cunoașterea arhitecturilor disponibile în prezent pe piață, a limbajelor de programare, a structurilor de date necesare precum și a bibliotecilor existente adaptate diverselor domenii precum prelucrarea imaginilor, computer vision, inteligența artificială, grafica 3D, calcule matematice, algebra.



Universitatea Națională de Știință și Tehnologie Politehnica București
Facultatea de Electronică, Telecomunicații și
Tehnologia Informației



Înțelegerea acestor aspecte împreună cu particularitățile fiecărei arhitecturi hardware creează premisele unor implementări eficiente, optimizate și adaptate cerințelor de calcul din ce în ce mai crescute odată cu creșterea complexității aplicațiilor. Prelucrarea rapidă a informației necesită utilizarea unor platforme adaptate, fie ca se utilizează arhitecturi cloud, arhitecturi GPU sau platforme mobile.

Data completării

Titular de curs

Titular(i) de aplicații

25.09.2025

S.I./Lect. Dr. George Valentin
STOICA

S.I./Lect. Dr. George Valentin
STOICA

Data avizării în departament

Director de departament

Conf.Dr.Ing. Bogdan Cristian FLOREA

Data aprobării în Consiliul
Facultății

Decan

26.09.2025

Prof. Dr. Mihnea Udrea