Universitatea Naţională de Ştiinţă şi Tehnologie Politehnica Bucureşti

Facultatea de Electronică, Telecomunicaţii şi

Tehnologia Informaţiei

FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituţia de învăţământ superior	**Universitatea Naţională de Ştiinţă şi Tehnologie Politehnica Bucureşti**
1.2 Facultatea	**Electronică, Telecomunicaţii şi Tehnologia Informaţiei**
1.3 Departamentul	**Dispozitive, Circuite şi Arhitecturi Electronice**
1.4 Domeniul de studii	Inginerie Electronică, Telecomunicaţii şi Tehnologii Informaţionale
1.5 Ciclul de studii	Masterat
1.6 Specializarea	Calcul Avansat în Sisteme Embedded

2. Date despre disciplină

2.1 Denumirea disciplinei (ro) (en)				Calcul paralel Parallel Computing			
2.2 Titularul activităţilor de curs				S.l./Lect. Dr. George Valentin STOICA			
2.3 Titularul activităţilor de seminar / laborator				S.l./Lect. Dr. George Valentin STOICA			
2.4 Anul de studiu	2	2.5 Semestrul	1	2.6. Tipul de evaluare	E	2.7 Regimul disciplinei	Ob
2.8 Tipul disciplinei		DA	2.9 Codul disciplinei	UPB.04.M3.O.26-24		2.10 Tipul de notare	Nota

3. Timpul total estimat (ore pe semestru al activităţilor 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 învăţământ	56	Din care: 3.5 curs	28	3.6 seminar/laborator	28
Distribuţia fondului de timp:					ore
Studiul după manual, suport de curs, bibliografie şi notiţe Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate Pregătire seminarii/ laboratoare/proiecte, teme, referate, portofolii şi eseuri					65
Tutorat					0
Examinări					4
Alte activităţi (dacă există):					0

3.7 Total ore studiu individual	69.00
3.8 Total ore pe semestru	125
3.9 Numărul de credite	5

4. Precondiţii (acolo unde este cazul)

4.1 de curriculum	Nu se plica
4.2 de rezultate ale învăţării	Nu se plica

5. Condiţii necesare pentru desfăşurarea optimă a activităţilor didactice (acolo unde este cazul)

5.1 Curs	Cursul se va desfasura intr-o sala dotata cu videoproiector.

5.2 Seminar/ Laborator/Proiect	Laboratorul va avea loc într -o sală dotată cu echipamente specifice, care trebuie să includă: PC -uri, mediu de dezvoltare software, cum ar fi MS Visual Studio, CPU multicore şi/sau GPU, cum ar fi NVIDIA GTX

6. Obiectiv general *(Se referă la intenţiile profesorilor pentru studenţi, la ceea ce studenţii vor fi învăţaţi în timpul cursului. Oferă o orientare cu privire la locul cursului în cadrul domeniului ştiinţific abordat, precum şi 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 învăţământ al specializării studiate etc.)*

Obiectivul cursului este ca studenţii să înţeleagă conceptele referitoare la arhitecturile de calcul de performanţă utilizate în domeniul tehnologiei informaţiei. Cursul prezintă sintetic limbajele folosite în platformele de procesare a informaţiilor precum C/C++/Java/Python/.Net şi continuă cu calculul paralel: arhitecturi multicore CPU şi GPU. A doua parte a cursului aprofundează arhitectura GPU şi mai precis arhitectura CUDA ca platformă de calcul de înaltă performanţă.

Aplicatiile de laborator au ca scop evidentierea conceptelor si notiunilor practice specifice calculului si prelucrarii informatiilor. Folosind dezvoltarea aplicaţiilor CUDA ca metode de experimentare sau folosind tehnologii precum OpenCL, studenţii pot experimenta singuri cunoştinţele dobândite în timpul cursului. Întregul set de noţiuni dobândite pe parcursul cursului poate fi experimentat în cadrul aplicaţiilor de laborator

7. Competenţe *(Capacitatea dovedită de a utiliza cunoştinţe, aptitudini şi abilităţi personale, sociale şi/sau metodologice în situaţii de muncă sau de studiu şi pentru dezvoltarea profesională şi personală. Reflectă cerinţele angajatorilor.)*

Specifice	După finalizarea acestui curs, studenţii dobândesc noţiunile de bază şi avansate de programare folosind arhitecturi GPU/CUDA. Aceste cunoştinţe combinate cu parcurgerea aplicaţiilor de laborator oferă posibilitatea de a dezvolta aplicaţii paralele de procesare a informaţiilor. De asemenea, studenţii vor fi capabili să dezvolte tehnici şi metode de programare, optimizare şi eficienţă a algoritmilor paraleli.
Transversale (generale)	Îndeplinirea sarcinilor profesionale cu identificarea exactă a obiectivelor de atins, a potenţialilor factori de risc, a resurselor disponibile, a aspectelor economico-financiare, a condiţiilor de îndeplinire a acestora, a etapelor de lucru, a timpului de lucru şi a termenelor aferente.

8. Rezultatele învăţării *(Sunt enunţuri sintetice referitoare la ceea ce un student va fi capabil să facă sau să demonstreze la finalizarea unui curs. Rezultatele învăţării reflectă realizările studentului şi mai puţin intenţiile profesorului. Rezultatele învăţării informează studenţii despre ceea ce se aşteaptă de la ei din punct de vedere al performanţei, pentru a obţine notele şi creditele dorite. Sunt definite în termeni concreţi, folosind verbe similare exemplelor de mai jos şi indică ceea ce se va urmări prin evaluare. Rezultatele învăţării vor fi astfel redactate încât să fie evidenţiată clar relaţia faţă de competenţele definite la punctul 7.)*

Cunoştinţe	*Rezultatul asimilării de informaţii prin învăţare. Cunoştinţele reprezintă ansamblul de fapte, principii, teorii şi practici legate de un anumit domeniu de muncă sau de studiu. Pot fi teoretice şi/sau faptice.* După finalizarea acestui curs, studenţii dobândesc noţiunile de bază şi avansate de programare folosind arhitecturi paralele, cum ar fi arhitecturi CPU multicore şi GPU/CUDA. Aceste cunoştinţe combinate cu parcurgerea aplicaţiilor practice oferă posibilitatea de a dezvolta aplicaţii eficiente de procesare a informaţiilor. De asemenea, studenţii vor fi capabili să dezvolte tehnici şi metode de programare, optimizare şi eficienţă a algoritmilor de procesare a informaţiilor.
Aptitudini	*Capacitatea de a aplica cunoştinţe şi de a utiliza know-how pentru a duce la îndeplinire sarcini şi a rezolva probleme. Aptitudinile sunt descrise ca fiind cognitive (implicând utilizarea gândirii logice, intuitive şi creative) sau practice (implicând dexteritate manuală şi utilizarea de metode, materiale, unelte şi instrumente).* Selectaţi şi grupaţi informaţii relevante într-un context dat.
Responsabilitate şi autonomie	*Capacitatea cursantului de a aplica în mod autonom şi responsabil cunoştinţele şi aptitudinile sale.* Demonstrează receptivitate la noi contexte de învăţare.

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 studenţilor la stabilirea propriului parcurs de învăţare, cum se identifică eventualele rămâneri în urmă şi ce măsuri remediale se adoptă în astfel de cazuri.)*

În cadrul acestei discipline, ţinând cont atât de orele de curs, cât şi de cele de aplicaţii, se vor utiliza atât metode de predare expozitive (prelecţie, expunere), cât şi conversaţional-interactiv, bazate pe modele de învăţare prin descoperire facilitate de explorarea directă şi indirectă a realităţii (experiment, demonstraţie, modelare), dar şi pe metode bazate pe acţiune, precum exerciţii, activităţi practice şi rezolvarea de probleme.

În activitatea didactică se vor folosi prelegeri, pe baza unor prezentări Power Point. Prezentările folosesc imagini şi diagrame astfel încât informaţiile prezentate să fie uşor de înţeles şi asimilat.

Prezentările pot fi realizate şi prin intermediul canalelor de comunicare online, pe platforma Teams.

Vor fi luate în considerare abilităţile de ascultare activă şi de comunicare asertivă, precum şi mecanismele de construcţie a feedback-ului. Abilităţile de lucru în echipă vor fi exersate pentru a rezolva diferite sarcini de învăţare.

10. Conţinuturi

CURS		
Capitolul	**Conţinutul**	**Nr. ore**
1	Introducere în calculul paralel 1.1. Paralelism motivant: aplicaţii intensive de calcul 1.2. Introducerea paralelismului în arhitecturile computerelor	2

2	Arhitecturi GPU 2.1. Arhitecturi GPU, GPGPU 2.2. Arhitectura CUDA	8
3	Arhitectura CUDA 3.1. Istorie, evoluţie 3.2. Descriere 3.3. Concepte specifice CUDA 3.4. Modele de programare CUDA 3.5. Modele de programare. Paralelism la nivel de date. Tipuri de memorie CUDA. 3.6. Aplicaţii. Analiza si optimizare. Consideraţii de performanţă.	10
4	Biblioteci bazate pe CUDA 4.1. Prezentarea bibliotecilor bazate pe arhitectura CUDA 4.2. cuBLAS, OpenACC, OpenCL, DNN	4
5	Aplicaţii 5.1. Aplicaţii de prelucrare a informaţiilor cu implementări GPU/CUDA: procesare imagini, inteligenţă artificială, viziune computerizată, grafică 3D	4
	Total:	28

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.	Conţinutul	Nr. ore
1	Biblioteci GPU/CUDA	2
2	Dezvoltarea unei aplicaţii CUDA	2
3	Dezvoltarea unei aplicaţii CUDA	2
4	Analiza/profilarea aplicatiilor CUDA	2
5	Optimizare	2
6	Aplicatii: procesare imagini	2
7	Aplicaţii: prelucrarea datelor	2
	Total:	

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	- cunoaşterea termenilor teoretici fundamentali; - cunostinte tehnici si arhitecturi de programare; - cunoasterea platformei GPU/CUDA; - cunoasterea unei biblioteci bazata pe GPU/CUDA	Test oral; subiectele acoperă întreaga materie, realizând o sinteză între cursul teoretic comparativ al materiei şi explicarea prin exerciţii a tehnicilor specifice programării platformelor GPU.	50%
11.5 Seminar/laborator/proiect	- cunoaşterea modului de proiectare a unui algoritm de analiză a imaginii pentru a rezolva o problemă dată; - cunoaşterea modului de transpunere a unui algoritm în cod; - demonstrarea funcţionării unui algoritm implementat.	Proiectul final de laborator, cuprinzând o componentă teoretică şi o componentă practică. Componenta teoretică se verifică prin întocmirea unui raport; componenta practică se evaluează prin verificarea modului de rezolvare (implementare, testare, operare) de către elev a unor probleme practice.	50%
11.6 Condiţii de promovare			
Proiectarea, implementarea şi demonstrarea de soluţii practice pentru procesarea imaginilor, viziunea computerizată, inteligenţa artificială folosind biblioteci accelerate CPU sau GPU/CUDA. Obţinerea a 50% din punctajul total. Obţinerea a 50% din punctajul aferent activităţii de laborator pe parcursul semestrului.			

12. Coroborarea conţinutului disciplinei cu aşteptările reprezentanţilor angajatorilor şi asociaţiilor profesionale reprezentative din domeniul aferent programului, precum şi cu stadiul actual al cunoaşterii în domeniul ştiinţific abordat şi practicile în instituţii de învăţământ superior din Spaţiul European al Învăţământului Superior (SEÎS)

Universitatea Naţională de Ştiinţă şi Tehnologie Politehnica Bucureşti

Facultatea de Electronică, Telecomunicaţii şi

Tehnologia Informaţiei

Prelucrarea 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 procesarea imaginilor, viziunea computerizată, inteligența artificială, grafica 3D, calculele matematice, algebra. Înțelegerea acestor aspecte împreună cu particularitățile fiecărei arhitecturi hardware creează premisele pentru implementări eficiente, optimizate și adaptate la cerințele de calcul din ce în ce mai mari odată cu complexitatea tot mai mare a aplicațiilor și a datelor acestora. Procesarea rapidă a informațiilor necesită utilizarea unor platforme adaptate, indiferent dacă sunt folosite arhitecturi cloud, arhitecturi GPU sau platforme mobile.

Data completării	Titular de curs	Titular(i) de aplicații
21.09.2026	S.l./Lect. Dr. George Valentin STOICA	S.l./Lect. Dr. George Valentin STOICA

Data avizării în departament	Director de departament
22.09.2026	Prof. Dr. Claudius Dan

Data aprobării în Consiliul Facultății	Decan
22.09.2026	Prof. Dr. Mihnea Udrea