**Universitatea Politehnica din Bucuresti**

**Facultatea de Inginerie in Limbi Straine**

**Departamentul de Inginerie in Limbi Straine**

**ANUNT**

**FISE DE DISCIPLINA**

**POSTURI DIDACTICE VACANTE**

**scoase la concurs în semestrul II, 2021 - 2022**

**PE PERIOADĂ NEDETERMINATĂ**

**Facultatea de inginerie IN LIMBI STRAINE**

***Departamentul de Inginerie in Limbi Straine***

**Semestrul 2 2021 - 2022**



**Annex no. 2**

**SUBJECT DESCRIPTION**

1. **Information about the program**

|  |  |
| --- | --- |
| * 1. Higher education institution | University POLITEHNICA of Bucharest |
| 1.2 Faculty | Faculty of Engineering in Foreign Languages |
| 1.3 Department | Department of Engineering in Foreign Languages |
| 1.4 Field of study | Computer science and information technology |
| 1.5 Study cycle | Undergraduate (license) |
| 1.6 Program / Qualification | Information Engineering / Engineer |

1. **Data about the subject**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 2.1 Name of subject | | | Advanced Computer Graphics | | | | |
| 2.2 Course holder | | |  | | | | |
| 2.3 Seminar holder | | |  | | | | |
| 2.4 Laboratory/project holder | | |  | | | | |
| 2.5 Year of study | 3 | 2.6 Semester | 5 | 2.7 Evaluation type | V | 2.8 Subject type | DD/DO |

1. **Estimated time (hours per semester) of didactic activities**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **3.1 Number of hours per week** | 3 | course hours | | | 1 | seminar | 0 | laboratory | 2 |
| **3.2. Number of hours per semester** | 42 | course hours | | | 14 | seminar | 0 | laboratory | 28 |
| **3.3.Distribution of spend time:** | | | | | | | |  |  |
| Study of textbooks, bibliography and course notes | | | | | | | | 22 | h. |
| Supplementary study in library, on electronic platforms, on the fieldwork | | | | | | | | 8 | h. |
| Preparation of seminars/laboratories, home assignments, papers, portfolios, essays | | | | | | | | 20 | h. |
| Tutoring | | | | | | | | 2 | h. |
| Examinations | | | | | | | | 10 | h. |
| Other activities | | | | | | | |  |  |
| **3.4 Total hours of individual study** | **62** | |  |
| **3.5 Total hours per semester [[1]](#footnote-1)** | **104** | |  |
| **3.6 Number of credits** | **4** | |  |

1. **Preconditions (where relevant)**

|  |  |
| --- | --- |
| 4.1 curriculum- related | * Data structures and algorithms, Programming Languages, Object Oriented Programming, Calculus, Linear algebra |
| 4.2 competence - related | * Basic competencies of geometry and trigonometry, C++/Java programming |

1. **Facilities and equipment (where relevant)**

|  |  |
| --- | --- |
| 5.1 for the course | * Video projector, blackboard or whiteboard |
| 5.2 for the course seminar | * - |
| 5.3 for the laboratory/project | * Computers with Visual Studio |

1. **Specific competences acquired**

|  |  |
| --- | --- |
| Professional competences | * Operating with the fundamental knowledge in electronics and computer science. * Ability to perform computations, proofs and applications for solving tasks specific to engineering, based on knowledge from fundamental and engineering sciences and from new information and communication technologies. * Design of hardware, software and communications components. |
| Transversal competences | * Demonstrating the spirit of initiative and action to update the professional, economic and organizational culture knowledge. |

1. **Course objectives (as resulting from the grid of specific competences)**

|  |  |
| --- | --- |
| 7.1 Subject general goal | * Studying the main algorithms for generating digital images, starting from the lowest level of the pixel, continuing with the generation of 2D polygons and reaching advanced notions of clipping, interpolation and approximation, generation of 3D objects, lighting and rendering effects. |
| 7.2 Specific objectives | • Ability to generate 2D objects (points, lines, polygons, polygonal and mathematical surfaces, fractals and other models)  • Ability to generate 3D surfaces (polygonal and mathematical)  • Ability to render using lights, fog and other effects  • Ability to optimize rendering by clipping |

1. **Content**

|  |  |  |
| --- | --- | --- |
| **8.1 Course** | **Teaching methods** | **Observations** |
| Introduction:  • Mathematical aspects: reference systems, points, lines, 2D and 3D vectors, operations with vectors, matrix, operations with matrix  • 2D transformations: homogeneous coordinates, translation, scaling, rotation, combining transformations  • 2D view: screen coordinates, point clipping, line clipping, area clipping | Face-to-face teaching assisted by video projector. | 4h |
| Rasterization:  • Simple and differential rasterization of lines  • Bresenham algorithm for lines and circles  • Filling polygons using Scanline | 2h |
| 3D viewing:  • 3D transformations: translation, scaling, rotation  • Perspective and parallel projections  • Perspective transformation  • 3D clipping: visual volume, point, line, polygon and object clipping | 4h |
| Representation of 3D objects:  • Quadratic surfaces  • Sweep  • Constructive solid geometry  • Based on octrees  • Based on fractals  • Curves and surfaces: Bezier, B-Spline, NURBS | 1h |
| • Rendering:  • Detection of visible surfaces: back-face detection, z-buffer, a-buffer, scan line, painter  • Lighting: ambient, diffuse, specular, types of light sources  • Lighting interpolation: flat, Gouraud, Phong  • Ray-casting, Ray-tracing | 3h |
| **Bibliography**   * Introduction to Computer Graphics”, J.D. Foley, A. van Dam, S.K. Feiner, J.F. Hughes & R.L. Phillips, Addison Wesley, 1997 * “Computer Graphics: Principles and Practice”, J.D. Foley, A. van Dam, S.K. Feiner & J.F. Hughes, Addison Wesley, 1995 – Foarte utilă pentru aspecte teoretice de bază. * „Fundamentals of Computer Graphics”, 4th Edition, Steve Marschner and Peter Shirley, A K Peters/CRC Press, 2015. * “3D Computer Graphics: A Mathematical Introduction with OpenGL”, Buss, Samuel R., 2003. ISBN: 9780521821032 * „Introduction to Computer Graphics”, David J. Eck, Hobart and William Smith Colleges, <https://math.hws.edu/eck/cs424/downloads/graphicsbook-linked.pdf> | | |
| **8.2 Seminar** | **Teaching methods** | **Observations** |
|  |  |  |
| **8.3 Laboratory** |  |  |
| * Working environment configuration (Visual Studio IDE) * Recap OOP concepts * Introduction to OpenGL * Using GLFW library for rendering windows * Double buffering | Individual and team activity, theoretical and practical.  Presentation of the basic notions needed for each laboratory on the blackboard and on the computer.  Homework evaluation. | First application is created which opens a GLFW window; the background color can be changed programmatically  2h |
| * Vertices and edges in OpenGL * Meshes; vbo, vao, ebo * Drawing points, lines and 2D primitives * First big homework is presented | Development of an application which draws 2D primitives in different positions on the screen.  2h |
| * Shaders in OpenGL (vertex shader, fragment shader) * OpenGL vectors, homogeneous coordinates * 2D transformations and corresponding transformation matrices:   + translation   + scaling to the origin and to a point   + rotation to the origin and to a point * Composing geometric transformations * GLM Library (OpenGL Mathematics) | An application is developed in which transformations (simple and composite) are applied to 2D objects. Testing the creation of a first animation (e.g. real-time rotation of a 2D object around the center of mass / around a fixed point)  2h |
| * Handling user input (polled input and event-based)   + Mouse events   + Keyboard events * Using callbacks for window interaction (closing, resizing) and for mouse interaction | An application is developed to test the two methods of managing input in GLFW. It adds functions to manipulate the graphics window and apply object transformations based on the input received.  2h |
| * Modifying the previous 2D architecture to work with 3D points and multiple faces per object * 3D transformations and corresponding transformation matrices:   + translation   + scaling to the origin and to a point   + rotation to the origin and to a point   • Perspective projection, MVP matrix (model-view-projection) | A program is developed that draws cubes which rotate on the screen and can be translated and scaled with the mouse.  2h |
| * Creating a camera in a graphic scene (position, viewing direction, “up” and “right” vectors) * View transforms obtained as a result of the translation and rotation of the camera on the 3 axes | A class is created responsible for creating and manipulating the transforms of a camera. Testing camera usage at user input in the 3D application from the previous lab.  2h |
| * Big homework 1 evaluation * Second big homework is presented | 2h |
| • Use of colors  • Textures - creation, mapping, texels, types  • Presentation of mipmaps  • Texturing parameters | An application is created in which different textures are loaded on 3D objects; testing texturing parameters; use of the mix function to interpolate textures.  2h |
| • Lighting  • Creating ambient, diffuse, specular light  • Phong, Gouraud models | The 3 lighting components are implemented (vertex shader and fragment shader level). A dynamic light source is added for better observation of light types.  2h |
| • Advanced OpenGL elements: face culling, cubemaps  • Advanced lighting - Phong improvements, shadows, depth map | Shadows are implemented in a 3D scene. A skybox is added using the cubemap concept.  2h |
| • Types of formats for 3D models; .obj model (Wavefront)  • Loading 3D objects (mesh loading) created in modeling tools in the OpenGL scene | Following a short modeling demo in a 3D software. Loading complex objects with their corresponding textures into the OpenGL scene.  2h |
| • Presentation of the game engine concept with classic examples (e.g. Unity, Unreal Engine)  • Presentation of a simple game engine created in OpenGL and including all the functionalities presented during the semester  • Creating relief elements by changing the coordinates at the vertex shader using the simple game engine shown.  • Unity bonus project presentation | Watching a short demo of using Unity3D as a game engine.  An OpenGL application is created in which mountains are simulated and at least two distinct pairs of shaders are used for various objects.  2h |
| • Introduction to virtual reality (VR) and augmented reality (RA)  • Presentation of VR devices (Google cardboard, Oculus rift, HTC Vive)  • Test open source AR applications (e.g. Google StreetView App, Expeditions) |  | Create simple AR applications using Google StreetView App, Expeditions  2h |
| • Big homework 2 and bonus project evaluation.  • Final laboratory marks. | 2h |
| **Bibliography**   * Same as mentioned for the course * “Computer Graphics with OpenGL”, D. Hearn & M. P. Baker, Prentice Hall, 2003 * OpenGL official docs: https://www.opengl.org/sdk/docs/man/ * Tutorial OpenGL online: <https://learnopengl.com/> * “3D Game Engine Architecture - Engineering Real-Time Applications with Wild Magic”, David H. Eberly, Magic Software, Inc., 2005 | | |

1. **Subject’s relevance to the epistemic community, professional associations and representative employers in fields significant for the program**

|  |
| --- |
| * Computer graphics are one of the most important fields of computer science, including areas such as computer aided design (CAD), scientific visualization, films, and games, virtual and augmented reality. * Computer graphics are playing a major role in industry, suppling the mathematical and the logical support of all fields mentioned previously. By following the advanced computer graphics course, students can learn the basics which can prove helpful for a future career in the gaming industry or in research based on virtual and augmented reality. |

1. **Assessment**

|  |  |  |  |
| --- | --- | --- | --- |
| Activity | 10.1 Evaluation criteria | 10.2 Evaluation methods | 10.3 Weight in final grade |
| 10.4 Course | Correctness of solving problems | Written examination | 40% |
| 10.5 Seminar |  |  |  |
| 10.6 Laboratory/Project | Laboratory activity | Evaluation of tasks given during laboratories | 30% |
| Homeworks | Evaluation of the 2 big homeworks given during the semester | 30% + 10% bonus  Final grade truncated to 100% |
| 10.7 Minimal standard of performance | | | |
| * 50% of the course grade * 50% of the lab grade | | | |

|  |  |  |
| --- | --- | --- |
| **Filling in date**  02.03.2022 | **Signature of the course holder**  Prof. Dr. ing. Costin-Anton Boiangiu | **Signature of the applications holder(s)**  Asist. Drd. ing. Iulia-Cristina Stănică |

|  |  |
| --- | --- |
| **Date of endorsement in the department** | **Signature of the Department Director** |
|  | Prof. Dr. Ing. George DRAGOI |

## ANEXA nr. 3 la metodologie

## FIŞA DISCIPLINEI

## 

1. **Date despre program**

|  |  |
| --- | --- |
| 1.1 Instituţia de învăţământ superior | **Universitatea Politehnica din Bucureşti** |
| 1.2 Facultatea / Departamentul | **Facultatea de Inginerie în Limbi Străine** |
| 1.3 Catedra | **Departamentul de Inginerie în Limbi Străine** |
| 1.4 Domeniul de studii | **Calculatoare şi tehnologia informaţiei** |
| 1.5 Ciclul de studii | **Licenţă** |
| 1.6 Programul de studii/Calificarea | **Ingineria informatiei / Inginer** |

1. **Date despre disciplină**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 2.1 Denumirea disciplinei | | | **Advanced Computer Graphics / Grafică computerizată avansată** | | | | | | |
| 2.2 Titularul activităţilor de curs | | | |  | | | | | |
| 2.3 Titularul activităţilor de laborator | | | |  | | | | | |
| 2.4 Anul de studiu | **3** | 2.5 Semestrul | | | 5 | 2.6 Tipul de evaluare | **V** | 2.7 Regimul disciplinei | **DD/DO** |

1. **Timpul total estimat (ore pe semestru) al activităţilor didactice**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **3.1 Număr de ore pe săptămână** | 3 | din care ore curs | | | 1 | seminar | 0 | laborator | 2 |
| **3.2. Numar ore pe semestru** | 42 | din care ore curs | | | 14 | seminar | 0 | laborator | 28 |
| **3.3.Distribuţia fondului de timp:** | | | | | | | |  |  |
| Studiul după manual, suport de curs, bibliografie şi notiţe | | | | | | | | 22 | h |
| Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate / pe teren | | | | | | | | 8 | h |
| Pregătire seminarii / laboratoare, teme, referate, portofolii şi eseuri | | | | | | | | 20 | h |
| Tutoriat | | | | | | | | 2 | h |
| Examinări | | | | | | | | 10 | h |
| Alte activităţi | | | | | | | |  |  |
| **3.4 Total ore studiu individual** | **62** | |  |
| **3.5 Total ore pe semestru [[2]](#footnote-2)** | **104** | |  |
| **3.6 Numărul de credite** | **4** | |  |

1. **Precondiţii** (acolo unde este cazul)

|  |  |
| --- | --- |
| 4.1 de curriculum | * Structuri de date și algoritmi, limbaje de programare, programare orientată pe obiecte, analiză matematică, algebră liniară |
| 4.2 de competenţe | * Competențe de bază de geometrie și trigonometrie, programare C++/Java |

1. **Condiţii** (acolo unde este cazul)

|  |  |
| --- | --- |
| 5.1. de desfăşurare a cursului | * Proiector video, tablă |
| 5.2. de desfășurare a seminarului | * Nu este cazul |
| 5.3. de desfășurare a laboratorului | * Laborator cu sisteme prevăzute cu Visual Studio |

1. **Competenţele specifice acumulate**

|  |  |
| --- | --- |
| **Competenţe profesionale** | * C1. Operarea cu fundamente ştiinţifice, ingineresti şi ale informaticii * C2. Proiectarea componentelor hardware, software şi de comunicaţii * C3. Soluţionarea problemelor folosind instrumentele ştiinţei şi ingineriei calculatoarelor |
| **Competenţe transversale** | * CT3. Demonstrarea spiritului de iniţiativă şi acţiune pentru actualizarea cunoştinţelor profesionale, economice şi de cultură organizaţională |

1. **Obiectivele disciplinei** (reieşind din grila competenţelor specifice acumulate)

|  |  |
| --- | --- |
| 7.1 Obiectivul general al disciplinei | * Studierea algoritmilor principali de generare de imagini digitale, c |
| 7.2 Obiectivele specifice | * Capacitatea de a genera obiecte 2D (puncte, linii, poligoane, suprafeţe poligonale şi matematice, fractali şi alte modele) * Capacitatea de a genera suprafeţe 3D (poligonale şi matematice) * Capacitatea de a randa folosind lumini, ceaţă şi alte efecte * Capacitatea de a optimiza randarea prin cliping |

1. **Conţinuturi**

|  |  |  |
| --- | --- | --- |
| 8. 1 Curs | Metode de predare | Observaţii |
| Introducere:   * Aspecte matematice: sisteme de referinţă, puncte, linii, vectori 2D şi 3D, operaţii cu vectori, matrice, operaţii cu matrice * Transformări 2D: coordonate omogene, translatare, scalare, rotaţie, combinarea transformărilor * Vizualizare în 2D: coordonate ecran, point clipping, line clipping, area clipping | Predare faţă în faţă asistată de videoproiector. | 4h |
| Rasterizarea:   * Rasterizarea simplă şi diferenţială a liniilor * Algoritmul Bresenham pentru linii şi cercuri * Umplerea poligoanelor folosind Scanline | 2h |
| Vizualizarea în 3D:   * Transformări 3D: translatare, scalare, rotaţie * Proiecţii perspective şi paralele * Transformarea perspectivă * Clipping în 3D: volumul vizual, point, line, polygon şi object clipping | 4h |
| Reprezentarea obiectelor 3D:   * Suprafeţe cuadrice * Sweep * Constructive solid geometry * Bazată pe arbori octali * Bazată pe fractali * Curbe şi suprafeţe: Bezier, B-Spline, NURBS | 1h |
| * Randarea: * Detecţia suprafeţelor vizibile: back-face detection, z-buffer, a-buffer, scan line, painter * Iluminare: ambientală, difuză, speculară, tipuri de surse de lumină * Interpolare iluminare: flat, Gouraud, Phong * Ray-casting, Ray-tracing | 3h |
| Bibliografie   * „Introduction to Computer Graphics”, J.D. Foley, A. van Dam, S.K. Feiner, J.F. Hughes & R.L. Phillips, Addison Wesley, 1997 * “Computer Graphics: Principles and Practice”, J.D. Foley, A. van Dam, S.K. Feiner & J.F. Hughes, Addison Wesley, 1995 – Foarte utilă pentru aspecte teoretice de bază. * „Fundamentals of Computer Graphics”, 4th Edition, Steve Marschner and Peter Shirley, A K Peters/CRC Press, 2015. * “3D Computer Graphics: A Mathematical Introduction with OpenGL”, Buss, Samuel R., 2003. ISBN: 9780521821032 * „Introduction to Computer Graphics”, David J. Eck, Hobart and William Smith Colleges, <https://math.hws.edu/eck/cs424/downloads/graphicsbook-linked.pdf> | | |
| 8. 2 Seminar/laborator | Metode de predare | Observaţii |
| * Configurare mediu de lucru (Visual Studio IDE) * Recapitulare concepte programare orientată pe obiecte * Introducere OpenGL * Utilizare GLFW pentru ferestre * Double Buffering | Activitate individuală și în echipă, teoretică şi practică.  Prezentarea noţiunilor necesare pentru fiecare laborator la tablă şi pe calculator.  Evaluarea temelor. | Se creează o primă aplicație ce deschide o fereastră GLFW; se setează culoarea de background  2 ore |
| * Vârfuri și muchii în OpenGL * Prezentarea conceptelor de mesh, vbo, vao, ibo * Desenarea de puncte, linii şi poligoane 2D * Se prezintă prima temă mare | Se dezvoltă o aplicație ce desenează primitive 2D în diferite poziții pe ecran.  2 ore |
| * Shadere în OpenGL (vertex shader, fragment shader) * Vectori în OpenGL, coordonate omogene * Transformări 2D și matricele de transformare corespunzătoare:   + translatare   + scalare faţă de origine şi faţă de un punct   + rotaţie faţă de origine şi faţă de un punct * Transformări geometrice compuse * Biblioteca GLM (OpenGL Mathematics) | Se dezvoltă o aplicație în care se aplică transformări (simple și compuse) obiectelor 2D. Se testează crearea unei prime animații (e.g. rotația în timp real a unui obiect 2D în jurul centrului de greutate / în jurul unui punct fix)  2 ore |
| * Interacțiunea cu utilizatorul (input de tip “polled” și bazat pe evenimente în GLFW)   + Folosirea mouse-ului pe post input   + Folosirea tastaturii pe post de input * Utilizarea funcțiilor de tip callback pentru interacțiunea cu ferastra (închidere, redimensionare) și pentru interacțiunea dintre mouse și obiectele de pe ecran | Se dezvoltă o aplicație în care se testează cele două metode de gestionare a input-ului în glfw. Se adaugă funcții de manipulare a ferestrei grafice și aplicarea de transformări obiectelor pe baza input-ului primit.  2 ore |
| * Modificarea arhitecturii 2D precedente pentru a putea lucra cu puncte 3D şi mai multe feţe per obiect * Transformări 3D și matricele de transformare corespunzătoare:   + translatare   + scalare faţă de origine şi faţă de un punct   + rotaţie faţă de origine şi faţă de un punct * Proiecţia perspectivă, matricea MVP (model-view-projection) | Se dezvoltă un program care desenează cuburi care se rotesc pe ecran şi care pot fi translatate şi scalate cu mouse-ul.  2 ore |
| * Crearea unei camere într-o scenă grafică (poziție, direcție de vizualizare, vector “sus” și vector “dreapta”) * Transformări de vizualizare obținute în urma translației și rotației camerei pe cele 3 axe | Se creează o clasă responsabilă pentru crearea și manipularea transformărilor unei camere. Testarea utilizării camerei la input-ul utilizatorului în aplicația 3D din laboratorul precedent.  2 ore |
| * Verificarea temei 1 * Se prezintă a doua temă mare | 2 ore |
| * Utilizarea culorilor în shadere * Texturi – creare, mapare, texeli, tipuri * Prezentare mipmaps * Parametri de texturare | Se creează o aplicație în care se încarcă diferite texturi pe obiecte 3D; testarea parametrilor de texturare; utilizarea funcției mix pentru interpolarea texturilor.  2 ore |
| * Iluminare * Crearea luminii ambientale, difuze, speculare * Modele Phong, Gouraud | Se implementează cele 3 componente de iluminare (vertex shader și fragment shader). Se adaugă o sursă de lumină dinamică pentru o mai bună observare a tipurilor de lumină.  2 ore |
| * Elemente avansate de OpenGL: face culling, cubemaps * Iluminare avansată – îmbunătățiri Phong, umbre, depth map | Se implementează umbre într-o scenă 3D. Se adaugă un skybox folosind conceptul de cubemap.  2 ore |
| * Tipuri de formate pentru modelele 3D; modelul .obj (Wavefront) * Încărcarea obiectelor 3D (mesh loading) create în tool-uri de modelare în scena OpenGL | Se urmărește un scurt demo de modelare într-un soft 3D. Se încarcă obiecte complexe cu texturile aferente în scena OpenGL.  2 ore |
| * Prezentarea conceptului de motor de joc (game engine) cu exemple clasice (ex. Unity, Unreal Engine) * Prezentarea unui game engine simplu creat în OpenGL și incluzând toate funcționalitățile prezentate de-a lungul semestrului * Crearea de elemente de relief prin modificarea coordonatelor la nivelul vertex shader-ului folosind motorul de joc simplu prezentat. * Prezentare proiect bonus Unity | Se urmărește un scurt demo al utilizării Unity3D ca motor de joc.  Se creează o aplicație în OpenGL în care se simulează munții și se utilizează minim două perechi distincte de shadere pentru obiecte diverse.  2 ore |
| * Introducere în realitatea virtuală (RV) și cea augmentată (RA) * Prezentare dispozitive RV (Google cardboard, Oculus rift, HTC Vive) * Testare aplicații open source RA (ex. Google StreetView App, Expeditions) | Crearea de aplicații RA simple folosind Google StreetView App, Expeditions  2 ore |
| * Verificarea temei 2 și a bonusului. * Încheierea situației de laborator. | 2 ore |
| Bibliografie   * Aceeaşi cu cea de la curs * “Computer Graphics with OpenGL”, D. Hearn & M. P. Baker, Prentice Hall, 2003 * OpenGL official docs: <https://www.opengl.org/sdk/docs/man/> * Tutorial OpenGL online: https://learnopengl.com/ | | |

1. **Coroborarea conţinuturilor disciplinei cu aşteptările reprezentanţilor comunităţii epistemice, asociaţiilor profesionale şi angajatori reprezentativi din domeniul aferent programului**

|  |
| --- |
| * Grafica computerizată este unul dintre cele mai importante ramuri ale informaticii, incluzând domenii precum proiectarea asistată de computer (CAD), vizualizarea științifică, filmele și jocurile, realitatea virtuală și augmentată. * Grafica computerizată joacă un rol major în industrie, asigurând suportul matematic și logistic al tuturor domeniilor menționate anterior. Urmând cursul de grafică avansată pe calculator, studenții pot învăța elementele de bază care se pot dovedi utile pentru o viitoare carieră în industria jocurilor sau în cercetarea bazată pe realitate virtuală și augmentată. |

1. **Evaluare**

|  |  |  |  |
| --- | --- | --- | --- |
| Tip activitate | 10.1 Criterii de evaluare | 10.2 Metode de evaluare | 10.3 Pondere din nota finală |
| 10.4 Curs | Corectitudinea rezolvării problemelor | Examen scris | 40% |
| 10.5 Seminar/laborator | Activitate Laborator | Evaluarea programelor ce trebuie dezvoltate în timpul laboratorului | 30% |
| Teme | Evaluarea temelor date pe parcursul semestrului | 30% + 10% bonus.  Notă finală trunchiată la 100 |
| 10.6 Standard minim de performanţă | | | |
| * 50% din punctajul cursului * 50% din punctajul laboratorului | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Data completării | Semnătura titularului de curs | | Semnătura titularului de seminar |
| 02.03.2022 |  | |  |
|  |  | |  |
| Data avizării în departament | | Semnătura directorului de departament | |
| ....................... | | Prof. Dr. Ing. George DRAGOI | |

**Annexe no. 2**

**FICHE DE LA DISCIPLINE**

1. **Informations sur le programme**

|  |  |
| --- | --- |
| * 1. Institut d’enseignement supérieur | Université POLITEHNICA de Bucarest |
| 1.2 Faculté | Faculté d’Ingénierie en Langues Etrangères |
| 1.3 Département | Département d’Ingénierie en Langues Etrangères |
| 1.4 Domaine d’études | Ordinateurs et technologie de l’information |
| 1.5 Cycle d’études | Licence |
| 1.6 Programme d’études / Qualification | Ingénierie de l'information/Ingénieur |

1. **Informations sur la discipline**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 2.1 Nom de la discipline | | | Réalité Virtuelle et Réalité Augmentée | | | | |
| 2.2 Titulaire de cours | | |  | | | | |
| 2.3 Titulaire pour les activités de séminaire | | |  | | | | |
| 2.4 Titulaire pour les activités du laboratoire/TP,s | | |  | | | | |
| 2.5 An d’études | 4 | 2.6 Semestre | 7 | 2.7 Type d’évaluation | E | 2.8 Régime (type) de la discipline | DS/DA |

1. **Temps total estimé (heures par semestre) pour les activités didactiques**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **3.1 Nombre d’heures par semaine** | 4 | dont heures de cours | | | 2 | projet | 1 | laboratoire | 1 |
| **3.2. Nombre d’heures par semestre** | 56 | dont heures de cours | | | 28 | projet | 14 | laboratoire | 14 |
| **3.3. Distribution du fond de temps:** | | | | | | | |  | **heures** |
| Etude d’après le manuel, support de cours, références | | | | | | | |  | 14 |
| Documentation supplémentaire dans les bibliothèques, sur les plateformes électroniques de spécialité/sur le terrain | | | | | | | |  | 12 |
| Préparation des séminaires/laboratoires, devoirs, rapports et essais. | | | | | | | |  | 14 |
| Tutorat | | | | | | | |  | 4 |
| Examinassions | | | | | | | |  | 4 |
| Autre activités | | | | | | | |  |  |
| **3.4 Total heures d’étude individuel** |  | | **48** |
| **3.5 Total Heures par semestre [[3]](#footnote-3)** |  | | **104** |
| **3.6 Nombre des crédits** |  | | **4** |

1. **Pré-requis (la ou il y a le cas)**

|  |  |
| --- | --- |
| 4.1 de curriculum | * Structures de données et algorithmes, programmation des ordinateurs et langages de programmation. |
| 4.2 de compétences | * Ensemble de compétences de base en programmation, géométrie, trigonométrie et physique |

1. **Termes (la ou il y a le cas)**

|  |  |
| --- | --- |
| 5.1 du déroulement du cours | * Salle de classe avec vidéoprojecteur |
| 5.2 du déroulement du séminaire | - |
| 5.3 du déroulement du laboratoire | * Salle de classe ayant des ordinateurs avec les logiciels nécessaires configurés |

1. **Compétences spécifiques acquises**

|  |  |
| --- | --- |
| Compétences professionnelles | * Effectuer des calculs, des démonstrations et des applications pour résoudre les tâches spécifiques d'ingénierie, basées sur les connaissances des sciences fondamentales, de l'ingénierie et des nouvelles technologies de l'information et des communications. * Développer des connaissances, des technologies numériques et des applications logicielles pour intégrer des produits intelligents et pour gérer la conformité aux normes de qualité et de sécurité des TIC. * Appliquer les politiques de l'entreprise, identifier les besoins des clients, mettre en œuvre la planification stratégique et effectuer des audits de qualité et de sécurité cybernétique. * Appliquer et développer des méthodologies de gestion de projet agiles en TIC tout au long du cycle de développement du système. |
| Compétences transversales | * L’analyse méthodique des problèmes rencontrés dans l’activité en identifiant les éléments pour lesquels il existe des solutions consacrées en assurant ainsi la réussite des charges professionnelles. * L’adaptation aux nouvelles technologies, le développement professionnel et personnel, par une formation continue en utilisant des documents imprimés, des logiciels spécialisés et des ressources électroniques dans la langue roumaine et dans une langue de circulation internationale. |

1. **Objectifs de la discipline (d’après la grille des compétences spécifiques accumulés)**

|  |  |
| --- | --- |
| 7.1 Objectif général de la discipline | * Le cours va initier les étudiants aux domaines émergents de la réalité virtuelle et réalité augmentée, en leur présentant les différents aspects de la réalité, les différentes catégories d'applications, les technologies et les dispositifs utilisés. * L'accent sera mis sur l'application des concepts dans la vie quotidienne pour résoudre des problèmes qui puissent survenir dans différents domaines (médecine, éducation, entreprises). * Les étudiants apprendront les concepts, les techniques et les outils nécessaires pour créer des applications de réalité virtuelle ou augmentée. |
| 7.2 Objectifs spécifiques | * Le cours fournira des études de cas réels, y compris des applications de réalité virtuelle ou augmentée qui ont eu du succès dans les différents domaines. * Les étudiants seront capables d'identifier les aspects interdisciplinaires impliqués dans les applications de réalité virtuelle ou de la réalité augmentée: éthiques, psychologiques, sociaux, économiques etc. * Les étudiants seront capables de créer des applications graphiques pour le desktop, Web ou mobile en utilisant les moteurs graphiques 3D connus (Unity, Unreal Engine). * Ils apprendront à intégrer des dispositifs de réalité virtuelle multimodaux dans leurs applications (Oculus Rift, HTC Vive, Google Cardboard, Leap Motion). * Les étudiants seront aussi capables d’intégrer le kit de développement Vuforia SDK dans des applications de réalité augmentée pour les appareils mobiles. * Ils passeront par toutes les étapes nécessaires au développement d’un jeu ou d’un système interdisciplinaire (médical, éducatif). |

1. **Contenus**

|  |  |  |
| --- | --- | --- |
| **8.1 Cours** | **Méthodes d’enseignement** | **Observations** |
| **1. Introduction à la réalité virtuelle et augmentée**  (définitions, concepts et composants spécifiques, statut actuel) | • Les méthodes d'exposition seront combinées avec des méthodes de conversation (conversation, débat)  • L’information sera présentée à travers des présentations PowerPoint sur le vidéoprojecteur  • Les présentations des étudiants pendant le cours seront encouragées, sur des sujets liés à la réalité virtuelle et augmentée | 2 heures |
| **2. Aspects physiologiques, psychologiques et philosophiques de la réalité virtuelle et de la réalité augmentée**  (simulation, réal vs virtuel, reproduction sensorielle, sens de la présence, interaction avec des environnements virtuels, immersion) | 2 heures |
| **3. Applications de la réalité virtuelle**  (jeux, applications MMO, communautés de socialisation et de simulation en ligne, applications militaires, simulateurs médicaux, applications d’éducation et de formation, simulations industrielles, de navigation et architecturales) | 2 heures |
| **4. Dispositifs multimodaux utilisés dans la réalité virtuelle**  • Dispositifs d’entrée: capteurs, détecteurs de mouvement, gants numériques etc.  • Périphériques de sortie: visuel, audio, haptique | 4 heures |
| **5. Techniques avancées dans la réalité virtuelle**  (principes de base de l’infographie, affichage stéréo, gestion de l’environnement virtuel à grande échelle, rendu en temps réel) | 4 heures |
| **6. Moteurs de jeu de réalité virtuelle**  • Les composants d'un moteur de jeu: géométrie, physique, animations, réseau, modélisation 3D, script, audio, interface utilisateur, etc.  • Exemples de moteurs de jeu: Unity, Unreal Engine, Cry Engine | 6 heures |
| **7. Logiciels de modélisation 3D**  (Maya, 3DS Max, Sketchup, Blender, Poser, fonctionnalités de base pour la création des modèles 3D dans Maya) | 2 heures |
| **8. La réalité augmentée - Définition, applications, technologies**  (définitions, différences entre la réalité augmentée et la réalité mixte, périphériques, applications dans différents domaines, réalité augmentée avec les appareils mobiles - Vuforia SDK) | 4 heures |
| **9. L'avenir de la réalité virtuelle et de la réalité augmentée**  (évolution, statut actuel, perspective de développement et possibilités d'utilisation dans de nouveaux domaines) | 2 heures |
| **Bibliographie (Références)**  1. Virtual Reality, Samuel Greengard, The MIT Press Essential Knowledge series  2. Learning Virtual Reality: Developing Immersive Experiences and Applications for Desktop, Web, and Mobile, Tony Parisi, Edition O'Reilly Media, 2015  3. Experience on Demand: What Virtual Reality Is, How It Works, and What It Can Do, Jeremy Bailenson, W. W. Norton & Company; 1st edition (27 Feb. 2018)  4. Practical Augmented Reality: A Guide to the Technologies, Applications and Human Factors for Ar and Vr, Steve Aukstakalnis, Addison-Wesley Professional; 1st edition (8 Sept. 2017)  5. Steven M. LaValle – Virtual Reality, Cambridge University Press, <http://lavalle.pl/vr/> | | |
| **8.2 Projet** | **Méthodes d’enseignement** | **Observations** |
| Développement en équipe d'une application de réalité virtuelle ou augmentée sous Unity. Une liste d'idées de projets issus de domaines variés, allant d'applications médicales, de formation ou d'applications de divertissement (jeux) sera fournie. |  | 14 heures |
| **8.3 Laboratoire** |  |  |
| **1. Introduction dans Unity**  (Premier projet dans Unity, conception et mise en œuvre, navigation, logique du jeu; création d’une application simple) | Démo, exercices | 2 heures |
| **2. Utilisation des Assets dans Unity**  (Modèles 3D, matériaux, textures, audio, animations) | Démo, exercices | 2 heures |
| **3. Plateformes d'applications dans Unity**  (desktop, Web, dispositifs mobiles, consoles) | Démo, exercices | 2 heures |
| **4. Gestion des périphériques d'entrée dans Unity**  (clavier, souris, écran tactile, manette de jeu etc.) | Démo, exercices | 2 heures |
| **5. Intégration des technologies de réalité virtuelle dans Unity**  (présentation de différents SDK, API, plugins pouvant être intégrés dans Unity, utilisation du dispositif HTC Vive/Oculus dans une application simple) | Démo, exercices | 2 heures |
| **6. Présentation des notions de base de réalité augmentée**  (présentation Adobe Aero, développement d'applications à l'aide d'outils spécialisés pour Desktop / mobile) | Démo, exercices | 2 heures |
| **7. Intégration des technologies de réalité augmentée dans Unity**  (Présentation AR Foundation, développement d'une application simple de réalité augmentée à partir d'un squelette de code fourni) | Démo, exercices | 2 heures |
| **Bibliographie**  1. Michelle Menard – Game Development with Unity, Edition Cengage Learning, 2011 | | |

1. **Collaboration des contenus de la discipline avec les attentes de la communauté épistémique, des associations professionnelles et des employeurs reconnus dans le domaine du programme**

|  |
| --- |
| * **Les exercices et les thèmes de projet proposés sont alignés sur les orientations de recherche et les exigences professionnelles actuelles dans les domaines de l’infographie, de la réalité virtuelle et de l'industrie des jeux vidéo.** |

1. **Evaluation**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Type d’activité | | 10.1 Critères d’évaluation | | | 10.2 Méthodes d’évaluation | | 10.3 part (pourcentage) de la note finale |
| 10.4 Cours | | Concepts enseignés au cours | | | Examen écrit | | 30% |
| Des présentations sur des thèmes actuels, liées à la réalité virtuelle ou augmentée | | | Examen oral | | 10% (extra) |
| 10.5 Projet | | Développement du projet final | | | Evaluation au parcours et finale du projet en équipe | | 50% |
| 10.6 Laboratoire/TP.s | | Activité au fil du semestre | | | Exercices pratiques au cours du laboratoire | | 20% |
| 10.7 Standard minime de performance | | | | | | | |
| * 50% des points de l’examen écrit du cours * 50% des points total du laboratoires (TP + projet) | | | | | | | |
| **Date d’élaboration**  02.03.2022 | | **Signature du titulaire du cours** | | | **Signature du/des titulaire/titulaires des applications** | |
| **Date de l’avis du département** | | | **Signature du directeur de Département** | | | |
|  | | | Prof. Dr. Ing. George DRAGOI | | | |

**Anexa nr. 2**

**FIŞA DISCIPLINEI**

1. **Date despre program**

|  |  |
| --- | --- |
| * 1. Instituţia de învăţământ superior | Universitatea POLITEHNICA din Bucureşti |
| 1.2 Facultatea | Facultatea de Inginerie în Limbi Străine |
| 1.3 Departamentul | Departamentul de Inginerie în Limbi Străine |
| 1.4 Domeniul de studii | Calculatoare si tehnologia informatiei |
| 1.5 Ciclul de studii | Licenta |
| 1.6 Programul de studii / Calificarea | Ingineria Informatiei/Inginer |

1. **Date despre disciplină**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 2.1 Denumire disciplina | | | Realitate virtuală și augmentată | | | | |
| 2.2 Titular activităţi de curs | | |  | | | | |
| 2.3 Titular activităţi de seminar | | | - | | | | |
| 2.4 Titular activităţi de laborator/lucrari | | |  | | | | |
| 2.5 Anul de studiu | 4 | 2.6 Semestrul | 7 | 2.7 Tipul de evaluare | E | 2.8 Regimul disciplinei | DS/DA |

1. **Timpul total estimat (ore pe semestru) al activităţilor didactice**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **3.1 Număr de ore pe săptămână** | 4 | din care ore curs | | | 2 | proiect | 1 | laborator | 1 |
| **3.2. Numar ore pe semestru** | 56 | din care ore curs | | | 28 | proiect | 14 | laborator | 14 |
| **3.3.Distribuţia fondului de timp:** | | | | | | | |  | **ore** |
| Studiul după manual, suport de curs, bibliografie şi notiţe | | | | | | | |  | 14 |
| Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate / pe teren | | | | | | | |  | 12 |
| Pregătire seminarii / laboratoare, teme, referate, portofolii şi eseuri | | | | | | | |  | 14 |
| Tutoriat | | | | | | | |  | 4 |
| Examinări | | | | | | | |  | 4 |
| Alte activităţi | | | | | | | |  |  |
| **3.4 Total ore studiu individual** |  | | **48** |
| **3.5 Total ore pe semestru [[4]](#footnote-4)** |  | | **104** |
| **3.6 Numărul de credite** |  | | **4** |

1. **Precondiţii (acolo unde este cazul)**

|  |  |
| --- | --- |
| 4.1 de curriculum | * Parcurgerea și/sau promovarea disciplinelor: structuri de date și algoritmi, programarea calculatoarelor și limbaje de programare. |
| 4.2 de competenţe | * Set de competențe de bază de programare, geometrie, trigonometrie și fizică |

1. **Condiţii (acolo unde este cazul)**

|  |  |  |
| --- | --- | --- |
| 5.1 de desfăşurare a cursului | * Sală de curs cu videoproiector |  |
| 5.2 de desfăşurare a seminarului | - |  |
| 5.3 de desfăşurare a laboratorului | * Sală cu un număr suficient de calculatoare, cu programele software necesare instalate |  |

1. **Competenţele specifice acumulate**

|  |  |
| --- | --- |
| Competenţe profesionale | * Efectuarea de calcule, demonstrații și aplicații, pentru rezolvarea de sarcini specifice ingineriei, pe baza cunoștințelor din științele fundamentale, inginerești și a noilor tehnologii informaționale și de comunicații. * Dezvoltarea de cunoștințe, tehnologii digitale și aplicații software pentru realizarea de produse inteligente integrate și gestionarea conformitații cu standardele de calitate și securitate în TIC. * Se ocupă de calitatea sistemelor TIC, stabilește obiective de asigurare a calității și gestionează conformitatea cu standardele de securitate în TIC. * Aplică și dezvoltă metodologii agile de management de proiect în TIC pe tot parcursul ciclului de viata al dezvoltării sistemelor |
| Competenţe transversale | * Analiza metodică a problemelor întâlnite în activitate, identificând elementele pentru care există soluţii consacrate, asigurând astfel îndeplinirea sarcinilor profesionale * Adaptarea la noile tehnologii, dezvoltarea profesională şi personală, prin formare continuă folosind surse de documentare tipărite, software specializat şi resurse electronice în limba română şi, cel puţin, într-o limbă de circulaţie internaţională |

1. **Obiectivele disciplinei (reieşind din grila competenţelor specifice acumulate)**

|  |  |
| --- | --- |
| 7.1 Obiectivul general al disciplinei | * Acest curs va introduce studenții în domeniul emergent al realității virtuale și augmentate, prezentându-le diferitele aspecte ale realității, categoriile de aplicații, tehnologiile și dispozitivele utilizate. * Accentul în tratatea subiectelor va fi pus pe aplicarea conceptelor în viața de zi cu zi, pentru rezolvarea problemelor din diferite domenii (medicină, educație, business) * Studenții vor învăța conceptele, tehnicile și instrumentele necesare pentru crearea de aplicații de realitate virtuală și augmentată |
| 7.2 Obiectivele specifice | * Cursul va oferi studii de caz reale, cuprinzând aplicații de succes de realitate virtuală și augmentată din diferite domenii. * Studenții vor fi capabili să identifice aspectele interdisciplinare implicate în aplicațiile de realitate virtuală și augmentată: etice, psihologice, sociale, economice etc. * Studenții vor fi capabili să realizeze aplicații grafice pentru desktop, web sau dispozitive mobile utilizând motoarele grafice 3D cunoscute (Unity, Unreal Engine) * Vor învăța să integreze dispozitive multimodale de realitate virtuală și augmentată în aplicațiile dezvoltate (Oculus Rift, HTC Vive, Google Cardboard, Leap Motion) * Studenții vor integra SDK-ul Vuforia pentru realizarea de aplicații de realitate augmentată pentru dispozitivele mobile. * Se vor parcurge toate etapele necesare dezvoltării unui joc sau a unui sistem complex interdisciplinar (medical, educațional) |

1. **Conţinuturi**

|  |  |  |
| --- | --- | --- |
| **8.1 Curs** | **Metode de predare** | **Observaţii** |
| 1. **Introducere în Realitatea Virtuală și Augmentată**   (definiții, concepte și componente specifice, stadiul actual) | * Se vor îmbina metode expozitive (prelegerea) cu cele conversative (conversații, dezbateri) * Expunerea informațiilor se va realiza cu ajutorul videoproiectorului, sub formă de prezentări PowerPoint * Vor fi încurajate și prezentări ale studenților la curs, pe teme preferate legate de subiectul cursului | 2 ore |
| 1. **Aspecte fiziologice, psihologice și filozofice ale Realității Virtuale și Augmentate**   (simulare, real vs virtual, reproduceri senzoriale, simțul prezenței, interacțiunea cu mediile virtuale, imersiunea) | 2 ore |
| 1. **Aplicații ale Realității Virtuale**   (jocuri, aplicații de tip MMO, comunități online de socializare și simulare, aplicații militare, simulatoare medicale, aplicații educative și pentru antrenament, simulări industriale, de navigație și arhitecturale) | 2 ore |
| 1. **Dispozitive multimodale utilizate în realitatea virtuală**  * dispozitive de intrare: senzori, detectori de mișcare, mănuși digitale etc. * dispozitive de ieșire: vizuale, audio, haptice | 4 ore |
| 1. **Tehnici Avansate în Realitatea Virtuală**   (fundamente de grafică pe calculator, principiul de stereo display, managementul mediilor virtuale la scară largă și randarea în timp real) | 4 ore |
| 1. **Motoare Grafice de Realitate Virtuală**  * Componentele unui motor grafic: geometrie, fizică, animații, networking, modelare 3D, scripting, audio, interfața cu utilizatorul etc. * Motoare cunoscute: Unity, Unreal Engine, Cry Engine | 6 ore |
| 1. **Instrumente Software de Modelare 3D**   (prezentare Maya, 3DS Max, Sketchup, Blender, Poser; funcționalități de bază pentru crearea de modele 3D în Maya) | 2 ore |
| 1. **Realitatea Augmentată – Definiție, Aplicații, Tehnologii**   (definiții, diferența dintre realitatea augmentată și realitatea mixtă, dispozitive periferice, aplicații în diverse domenii, realitatea augmentată cu dispozitivele mobile – Vuforia SDK) | 4 ore |
| 1. **Viitorul Realității Virtuale și Augmentate**   (evoluția până în prezent, stadiul actual, perspective de dezvoltare și posibilități de extindere către noi domenii) | 2 ore |
| **Bibliografie**  1. Virtual Reality, Samuel Greengard, The MIT Press Essential Knowledge series  2. Learning Virtual Reality: Developing Immersive Experiences and Applications for Desktop, Web, and Mobile, Tony Parisi, Edition O'Reilly Media, 2015  3. Experience on Demand: What Virtual Reality Is, How It Works, and What It Can Do, Jeremy Bailenson, W. W. Norton & Company; 1st edition (27 Feb. 2018)  4. Practical Augmented Reality: A Guide to the Technologies, Applications and Human Factors for Ar and Vr, Steve Aukstakalnis, Addison-Wesley Professional; 1st edition (8 Sept. 2017)  5. Steven M. LaValle – Virtual Reality, Cambridge University Press, <http://lavalle.pl/vr/> | | |
| **8.2 Proiect** | **Metode de predare** | **Observaţii** |
| Dezvoltarea în echipe a unei aplicații de realitate virtuală sau augmentată, utilizând Unity. Se va oferi o listă de idei de proiecte din domenii diverse, de la aplicații medicale, aplicații de training sau de divertisment (jocuri). |  | 14 ore |
| **8.3 Laborator** |  |  |
| 1. **Introducere în Unity**   (primul proiect în Unity, design-ul unui joc, navigarea, logica jocului; crearea de aplicații simple) | Demo, exerciții | 2 ore |
| 1. **Utilizarea de Assets în Unity**   (modele 3D, materiale, texturi, audio, animații) | Demo, exerciții | 2 ore |
| 1. **Platforme de Aplicații în Unity**   (desktop, web, dispozitive mobile, console) | Demo, exerciții | 2 ore |
| 1. **Gestionarea Dispozitivelor de Intrare în Unity**   (tastatură, mouse, ecran tactil, joystick etc.) | Demo, exerciții | 2 ore |
| 1. **Integrarea Tehnologiilor de Realitate Virtuală în Unity**   (prezentarea diferitelor SDK-uri, API-uri, pluginuri ce pot fi integrate în Unity; utilizarea dispozitivelor HTC Vive/Oculus cu o aplicație simplă) | Demo, exerciții | 1. ore |
| **6. Prezentarea notiunilor de baza de Realitate Augmentata**  (prezentarea Adobe Aero, dezvoltarea aplicatiilor folosind tool-uri specializate pentru Desktop / mobile) | Demo, exerciții | 2 ore |
| **7. Integrarea Tehnologiilor de Realitate Augmentată în Unity**  (prezentarea AR Foundation, dezvoltarea unei aplicatii simple de realitate augmentata pornind de la un schelet furnizat) | Demo, exerciții | 2 ore |
| **Bibliografie**   1. Michelle Menard – Game Development with Unity, Editura Cengage Learning, 2011 | | |

1. **Colaborarea conţinuturilor disciplinei cu aşteptările reprezentanţilor comunităţii epistemice, asociaţiilor profesionale şi angajatori reprezentativi din domeniul aferent programului**

|  |
| --- |
| * **Exercițiile propuse și temele de proiect sunt aliniate direcțiilor de cercetare și cerințelor profesionale actuale din domeniile aferente graficii, realității virtuale și industriei jocurilor pe calculator.** |

1. **Evaluare**

|  |  |  |  |
| --- | --- | --- | --- |
| Tip activitate | 10.1 Criterii de evaluare | 10.2 Metode de evaluare | 10.3 Pondere din nota finală |
| 10.4 Curs | Concepte predate la curs | Examen scris | 30% |
| Realizarea de prezentări pe teme actuale, de impact, din domeniul realității virtuale sau augmentate | Evaluare orală | 10% (suplimentare) |
| 10.5 Proiect | Dezvoltarea proiectului final | Evaluarea pe parcurs și finală a proiectului pe echipe | 50% |
| 10.6 Laborator/lucrari | Activitatea pe parcurs | Exerciții practice în timpul laboratorului | 20% |
| 10.7 Standard minim de performanţă | | | |
| * 50% din punctajul examenului scris de la curs * 50% din punctajul total laborator+proiect | | | |

|  |  |  |
| --- | --- | --- |
| **Data completării**  02.03.2022 | **Semnătura titularului de curs** | **Semnătura titularului/ titularilor de aplicaţii** |

|  |  |
| --- | --- |
| **Data avizării în departament** | **Semnătura directorului de Departament** |
|  | Prof. Dr. Ing. George DRAGOI |

**Annexe no. 2**

**FICHE DE LA DISCIPLINE**

1. **Informations sur le programme**

|  |  |
| --- | --- |
| * 1. Institut d’enseignement supérieur | Université POLITEHNICA de Bucarest |
| 1.2 Faculté | Faculté d’Ingénierie en Langues Etrangères |
| 1.3 Département | Département d’Ingénierie en Langues Etrangères |
| 1.4 Domaine d’études | Ordinateurs et technologie de l’information |
| 1.5 Cycle d’études | Licence |
| 1.6 Programme d’études / Qualification | Ingénierie de l’information/ingénieur |

1. **Informations sur la discipline**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 2.1 Nom de la discipline | | | Ingénierie des systèmes | | | | |
| 2.2 Titulaire de cours | | |  | | | | |
| 2.3 Titulaire pour les activités de séminaire | | |  | | | | |
| 2.4 Titulaire pour les activités du laboratoire/TP,s | | |  | | | | |
| 2.5 An d’études | 4 | 2.6 Semestre | 8 | 2.7 Type d’évaluation | V | 2.8 Régime (type) de la discipline | DS/DA |

1. **Temps total estimé (heures par semestre) pour les activités didactiques**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **3.1 Nombre d’heures par semaine** | 3 | dont heures de cours | | | 2 | seminaire |  | laboratoire | 1 |
| **3.2. Nombre d’heures par semestre** | 42 | dont heures de cours | | | 28 | seminaire |  | laboratoire | 14 |
| **3.3. Distribution du fond de temps:** | | | | | | | |  | **heures** |
| Etude d’après le manuel, support de cours, références | | | | | | | |  | 7 |
| Documentation supplémentaire dans les bibliothèques, sur les plateformes électroniques de spécialité/sur le terrain | | | | | | | |  | 5 |
| Préparation des séminaires/laboratoires, devoirs, rapports et essais. | | | | | | | |  | 15 |
| Tutoriat | | | | | | | |  | 6 |
| Examinations | | | | | | | |  | 3 |
| Autre activités | | | | | | | |  |  |
| **3.4 Total heures d’étude individuel** |  | | **36** |
| **3.5 Total Heures par semestre [[5]](#footnote-5)** |  | | **78** |
| **3.6 Nombre des crédits** |  | | **2** |

1. **Prerequis (la ou il y a le cas)**

|  |  |
| --- | --- |
| 4.1 de curriculum | * Participation et / ou la promotion des sujets suivants: Méthodes de développement de logiciels |
| 4.2 de compétences |  |

1. **Termes (la ou il y a le cas)**

|  |  |
| --- | --- |
| 5.1 du déroulement du cours | * Salle de conférences avec vidéoprojecteur |
| 5.2 du déroulement du séminaire |  |
| 5.3 du déroulement du laboratoire | * Salle d'ordinateur et connexion internet |

1. **Compétences spécifiques acquises**

|  |  |
| --- | --- |
| Compétences professionnelles | * Travailler avec fondements de mathématique, de ‘ingénierie et de l’informatique. * Conception des composants matériels, logiciels et de communication. * Trouver des solutions en utilisant les outils de la science et de l’ingénierie des ordinateurs. * Amélioration des performances des systèmes matériels, logiciels et de communication. * Conception et intégration des systèmes informatiques en utilisant des technologies et d’environnements de programmation. * La conception, la gestion du cycle de vie, l’intégration et l’intégrité des systèmes matériels, logiciels et de communication. . * La maintenance et l’exploitation des systèmes matériels, logiciels et de communication. * La conception et l’utilisation des systèmes intelligents.   L’ingénierie de la programmation (génie du logiciels) |
| Compétences transversales | * Le comportement honorable, responsable, étique, dans l’esprit des lois, pour assurer la réputation scientifique. * L’identification, la description et le déroulement des processus du management des projets, en prenant les différents rôles dans une équipe et la description, claire et concise, dans une langue de circulation internationales, des résultats du domaine d’activité.   La démonstration de l’esprit d’équipe et l’action pour la mise à jour des connaissances professionnelles, économiques et de culture organisationnelle. |

1. **Objectifs de la discipline (d’après la grille des conpetences spécifiques accumulés)**

|  |  |
| --- | --- |
| 7.1 Objectif général de la discipline | * définition des systemes complexe sur la nécessité d'appliquer les principes de l'ingénierie des systèmes pour développer de tels systèmes complexes; fournir des informations sur les modèles les plus populaires des cycles de vie des systèmes et des stades de développement des systèmes (phase de développement de phase de développement conceptuel lui-même et post-développement), fournissant des informations sur relation entre les systèmes d'ingénierie et d'autres disciplines telles que la gestion de projet, gestion des risques et la gestion de la qualité |
| 7.2 Objectifs spécifiques | * • présenter des études de cas réels, fournissant des informations sur les outils utilisés dans les systèmes d'ingénierie (tels que MS Project, plan de risque, le plan de la qualité, des outils qui utilisent SysML - Modelio, paradigme visuel) * • SysML diagrammes de modélisation des systèmes * • Développer des diagrammes de Gantt et les sites WBS * • l'élaboration d'une proposition de projet pour une équipe de projet complexe |

1. **Contenus**

|  |  |  |
| --- | --- | --- |
| **8.1 Cours** | **Méthodes d’enseignement** | **Observations** |
| Introduction a l'ingénierie des systèmes (principes, approches, activités et documents domaines connexes) | Conférence | 2 heures |
| Le processus de développement des systèmes (modèles de systèmes de LifeCycles caractéristiques d'évolution du processus de développement, les méthodes d'ingénierie des systèmes, l'ingénierie des systèmes et la gestion des projets, des graphiques et WBS Gantt) | Conférence | 2 heures |
| La gestion des risques dans l'ingénierie des systèmes (risque et l'effort tout au long du cycle de vie d'un risque de système - définition, les composants, les sources et les réponses, les méthodes de modélisation) | Conférence | 2 heures |
| Phase de développement conceptuelle des systèmes (analyse des besoins, analyse des spécifications, l'efficacité opérationnelle du modèle, en explorant les concepts, les opérations, l'analyse des alternatives, la définition des concepts) | Conférence | 2 heures |
| Systèmes d'ingénierie à base de la modélisation (spécification du langage SysML - schémas de structure, les exigences et le comportement) | Conférence | 8 heures |
| Phase de développement appropriée des systèmes (développement avancé, la conception technique, l'intégration et l'évaluation | Conférence | 2 heures |
| Post-développement (production, exploitation et support) | Conférence | 2 heures |
| Gestion de la qualité (modèles de qualité, l'assurance de la qualité du défaut / panne / erreur, classification des défauts, l'analyse des causes des défauts, comme avant et après produit de retrait sur le marché découverte intelligente des défauts, la qualité des produits par rapport à la qualité des modèles pour les processus de qualité, les mesures informations de mesure de qualité et modèles) | Conférence | 4 heures |
| Ingénierie des systèmes logiciels (appliqué les principes d'ingénierie des systèmes dans le développement de logiciels, les activités systèmes logiciels d'ingénierie) | Conférence | 2 heures |
| **Bibliographie (Références)**  1. ISO/IEC 15288:2008(E), IEEE Std 15288-2008, Second edition, 2008-02-01 Systems and software engineering — System life cycle processes, (Ingénierie des systèmes et du logiciel — Processus du cycle de vie du système) <https://www.iso.org/obp/ui/fr/#iso:std:iso-iec:15288:ed-2:v1:en>  2. Kossiakoff, A., Sweet, W.N., Seymour, A.J., Biemer, S.M. - Systems Engineering Principles and Practice, 2nd edition, Wiley Series in Systems Engineering and Management, 2003  3. Friedenthal, S., Moore, A., Steiner, R.A - Practical Guide to SysML, 2nd edition, The MK/OMG Press, 2011  4. Blanchard, B.S., Fabrycky W.J. - Systems Engineering and Analysis, 5th edition, Prentice Hall, 2010  5. Constanta-Nicoleta Bodea, Maria-Iuliana Dascalu - IT Risk Evaluation Model Using Risk Maps and Fuzzy Inference, IJITPM 1(2), pp. 79-97, 2010 | | |
| **8.2 Séminaire** | **Méthodes d’enseignement** | **Observations** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| **8.3 Laboratoire** |  |  |
| Études de cas d’introduction | Exercices de démonstration, exercices | 2 heures |
| Les instruments utilisés dans l'ingénierie des systèmes - MS Project | Exercices de démonstration, exercices | 2 heures |
| Les instruments utilisés dans l'ingénierie des systèmes - matrice des risques | Exercices de démonstration, exercices | 2 heures |
| SysML | Exercices de démonstration, exercices | 8 heures |
| **Bibliographie**  1**.** Meme que pour le cours  2. Friedenthal, S., Moore, A., Steiner, R.A - Practical Guide to SysML, 2nd edition, The MK/OMG Press, 2011 | | |

1. **Collaboration des contenus de la discipline avec les attentes de la communauté épistémique, des associations professionnelles et des employeurs reconnus dans le domaine du programme**

|  |
| --- |
| * Les exercices proposés sont alignés avec des questions d'entrevues professionnelles et les orientations de recherche en cours. |

1. **Evaluation**

|  |  |  |  |
| --- | --- | --- | --- |
| Type d’activité | 10.1 Critères d’évaluation | 10.2 Méthodes d’évaluation | 10.3 part (pourcentage) de la note finale |
| 10.4 Cours | Les concepts enseignés en classe | épreuve écrite | 20% |
| Capacité d'appliquer les concepts enseignés en classe par des exercices de résolution | épreuve écrite | 10% |
| 10.5 Séminaire |  |  |  |
|  |  |  |
| 10.6 Laboratoire/TP.s | L'activité au cours du semestre | Exercices pratiques en laboratoire | 20% |
| Résolution des problèmes pratiques | Devoirs | 10% |
| Résolution des problèmes pratiques | Projet | 40% |
| 10.7 Standard minime de performance | | | |
| * **Au moins 4,5 sur 10 points; Laboratoire / projet minimum 50% de points; au moins 50% des points attribués pour l'ensemble des activités** | | | |

|  |  |  |
| --- | --- | --- |
| **Date d’élaboration** | **Signature du titulaire du cours** | **Signature du/des titulaire/titulaires des applications** |

|  |  |
| --- | --- |
| **Date de l’avis du département** | **Signature du directeur de Département** |
|  | Prof. Dr. Ing. George DRAGOI |

**Anexa nr. 2**

**FIŞA DISCIPLINEI**

1. **Date despre program**

|  |  |
| --- | --- |
| * 1. Instituţia de învăţământ superior | Universitatea POLITEHNICA din Bucureşti |
| 1.2 Facultatea | Facultatea de Inginerie în Limbi Străine |
| 1.3 Departamentul | Departamentul de Inginerie în Limbi Străine |
| 1.4 Domeniul de studii | Calculatore si tehnologia informatiei |
| 1.5 Ciclul de studii | licenta |
| 1.6 Programul de studii / Calificarea | Ingineria informatiei/inginer |

1. **Date despre disciplină**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 2.1 Denumire disciplina | | | Ingineria sistemelor | | | | |
| 2.2 Titular activităţi de curs | | |  | | | | |
| 2.3 Titular activităţi de seminar | | |  | | | | |
| 2.4 Titular activităţi de laborator/lucrari | | |  | | | | |
| 2.5 Anul de studiu | 4 | 2.6 Semestrul | 8 | 2.7 Tipul de evaluare | V | 2.8 Regimul disciplinei | DS/DA |

1. **Timpul total estimat (ore pe semestru) al activităţilor didactice**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **3.1 Număr de ore pe săptămână** | 3 | din care ore curs | | | 2 | seminar |  | laborator | 1 |
| **3.2. Numar ore pe semestru** | 42 | din care ore curs | | | 28 | seminar |  | laborator | 14 |
| **3.3.Distribuţia fondului de timp:** | | | | | | | |  | **ore** |
| Studiul după manual, suport de curs, bibliografie şi notiţe | | | | | | | |  | 7 |
| Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate / pe teren | | | | | | | |  | 5 |
| Pregătire seminarii / laboratoare, teme, referate, portofolii şi eseuri | | | | | | | |  | 15 |
| Tutoriat | | | | | | | |  | 6 |
| Examinări | | | | | | | |  | 3 |
| Alte activităţi | | | | | | | |  |  |
| **3.4 Total ore studiu individual** |  | | **36** |
| **3.5 Total ore pe semestru [[6]](#footnote-6)** |  | | **78** |
| **3.6 Numărul de credite** |  | | **2** |

1. **Precondiţii (acolo unde este cazul)**

|  |  |
| --- | --- |
| 4.1 de curriculum | * Frecventarea si/sau promovarea urmatoarelor discipline: Metode de dezvoltare a programelor |
| 4.2 de competenţe |  |

1. **Condiţii (acolo unde este cazul)**

|  |  |
| --- | --- |
| 5.1 de desfăşurare a cursului | * Sala cu videoproiector |
| 5.2 de desfăşurare a seminarului |  |
| 5.3 de desfăşurare a laboratorului | * Sala cu calculatoare si conexiune la internet |

1. **Competenţele specifice acumulate**

|  |  |
| --- | --- |
| Competenţe profesionale | * Operarea cu fundamentele matematice, inginereşti şi ale informaticii. * Proiectarea componentelor hardware, software şi de comunicaţii. * Soluţionarea problemelor folosind instrumentele ştiinţei şi ingineriei calculatoarelor. * Îmbunătăţirea performanţelor sistemelor hardware, software şi de comunicaţii. * Proiectarea şi integrarea sistemelor informatice utilizând tehnologii şi medii de programare. * Proiectarea, gestionarea ciclului de viaţă, integrarea şi integritatea sistemelor hardware, software şi de comunicaţii. * Întreţinerea şi exploatarea sistemelor hardware, software şi de comunicaţii. * Proiectarea si utilizarea sistemelor inteligente. * Ingineria programarii |
| Competenţe transversale | * Comportarea onorabilă, responsabilă, etică, în spiritul legii pentru a asigura reputaţia profesiei. * Identificarea, descrierea şi derularea proceselor din managementul proiectelor, cu preluarea diferitelor roluri în echipă şi descrierea clară şi concisă, verbal şi în scris, într-o limbă de circulaţie internaţională, a rezultatelor din domeniul de activitate. * Demonstrarea spiritului de iniţiativă şi acţiune pentru actualizarea cunoştinţelor profesionale, economice şi de cultură organizaţională |

1. **Obiectivele disciplinei (reieşind din grila competenţelor specifice acumulate)**

|  |  |
| --- | --- |
| 7.1 Obiectivul general al disciplinei | * definirea sistemelor complexe si evidentierea nevoii de a aplica principii din ingineria sistemelor pentru a dezvolta astfel de sisteme complexe; oferirea de informatii cu privire la cele mai cunoscute modele de cicluri de viata ale sistemelor si la fazele procesului de dezvoltare a sistemelor (faza de dezvoltare conceptuala, faza de dezvoltare propriu-zisa si faza de post-dezvoltare), oferirea de informatii cu privire la relatia dintre ingineria sistemelor si alte discipline, cum ar fi managementul proiectelor, managementul riscului si managementul calitatii |
| 7.2 Obiectivele specifice | * prezentarea unor studii de caz reale, oferirea de informatii cu privire la instrumentele folosite in ingineria sistemelor (cum ar fi Ms Project, planul de risc, planul de calitate, instrumente ce utilizeaza SysML – Modelio, Visual Paradigm) * modelarea sistemelor prin diagrame SysML * dezvoltarea de grafice Gantt si de WBS-uri * dezvoltarea unei propuneri de proiect pentru un proiect complex, in echipa |

1. **Conţinuturi**

|  |  |  |
| --- | --- | --- |
| **8.1 Curs** | **Metode de predare** | **Observaţii** |
| **Introducere in ingineria sistemelor** (Principii, abordari, activitati si documente, domenii conexe) | Prelegere | 2 ore |
| **Procesul dezvoltarii sistemelor** (modele pentru ciclurile de viata ale sistemelor, caracteristicile evolutive ale procesului de dezvoltare, metode din ingineria sistemelor, ingineria sistemelor si managementul proiectelor, WBS si grafice Gantt) | Prelegere | 4 ore |
| **Managementul riscurilor in ingineria sistemelor** (risc si efort pe parcursul ciclului de viata al unui sistem, riscuri – definitie, componente, surse, raspunsuri , metode de modelare) | Prelegere | 2 ore |
| **Etapa de dezvoltare conceptuala a sistemelor** (analiza nevoilor, analiza specificatiilor, modelul eficacitatii operationale, explorarea conceptelor, operatiunile, analiza alternativelor, definitia conceptelor) | Prelegere | 2 ore |
| **Ingineria sistemelor bazate pe modelare** (specificatiile limbajului SysML – diagrame structurale, ale cerintelor si de comportament) | Prelegere | 8 ore |
| **Etapa dezvoltarii propriu-zise a sistemelor** (dezvoltare avansata, design-ul ingineresc, integrarea si evaluarea | Prelegere | 2 ore |
| **Etapa post-dezvoltare** (Productie, operatii si suport) | Prelegere | 2 ore |
| **Managementul calitatii** (modele de calitate, planul asigurarii calitatii, defect/esec/eroare, clasificarea defectelor, analiza cauzelor defectelor, calitate inainte si dupa scoaterea pe piata a produsului inteligent, descoperirea defectelor, calitatea produsului versus calitatea procesului, modele pentru calitatea proceselor, metrici de calitate si modele de masurare a informatiei) | Prelegere | 4 ore |
| **Ingineria sistemelor software** (principiile ingineriei sistemelor aplicabile in dezvoltarea de software, activitatile ingineriei sistemelor software) | Prelegere | 2 ore |
| **Bibliografie**  1. ISO/IEC 15288:2008(E), IEEE Std 15288-2008, Second edition, 2008-02-01 Systems and software engineering — System life cycle processes, (Ingénierie des systèmes et du logiciel — Processus du cycle de vie du système) <https://www.iso.org/obp/ui/fr/#iso:std:iso-iec:15288:ed-2:v1:en>  2. Kossiakoff, A., Sweet, W.N., Seymour, A.J., Biemer, S.M. - Systems Engineering Principles and Practice, 2nd edition, Wiley Series in Systems Engineering and Management, 2003  3. Friedenthal, S., Moore, A., Steiner, R.A - Practical Guide to SysML, 2nd edition, The MK/OMG Press, 2011  4. Blanchard, B.S., Fabrycky W.J. - Systems Engineering and Analysis, 5th edition, Prentice Hall, 2010  5. Constanta-Nicoleta Bodea, Maria-Iuliana Dascalu - IT Risk Evaluation Model Using Risk Maps and Fuzzy Inference, IJITPM 1(2), pp. 79-97, 2010 | | |
| **8.2 Seminar** | **Metode de predare** | **Observaţii** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| **8.3 Laborator** |  |  |
| Studii de caz introductive | Demo, exercitii | 2 ore |
| Instrumente folosite in ingineria sistemelor – MS Project | Demo, exercitii | 2 ore |
| Instrumente folosite in ingineria sistemelor – matricea riscurilor | Demo, exercitii | 2 ore |
| SysML | Demo, exercitii | 8 ore |
| **Bibliografie**  1**.** Aceeasi mentionata la curs  2. Friedenthal, S., Moore, A., Steiner, R.A - Practical Guide to SysML, 2nd edition, The MK/OMG Press, 2011 | | |

1. **Colaborarea conţinuturilor disciplinei cu aşteptările reprezentanţilor comunităţii epistemice, asociaţiilor profesionale şi angajatori reprezentativi din domeniul aferent programului**

|  |
| --- |
| * **Exercitiile propuse sunt aliniate intrebarilor din interviurile profesionale si directiilor de cercetare actuale.** |

1. **Evaluare**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Tip activitate | | 10.1 Criterii de evaluare | | | 10.2 Metode de evaluare | | 10.3 Pondere din nota finală |
| 10.4 Curs | | Conceptele predate la curs | | | Examen scris | | 20% |
| Capacitatea de aplicare a conceptelor predate la curs prin rezolvarea exercitiilor | | | Examen scris | | 10% |
| 10.5 Seminar | |  | | |  | |  |
|  | | |  | |  |
| 10.6 Laborator/lucrari | | Activitatea pe parcurs | | | Exercitii practice in timpul laboratorului | | 20% |
| Rezolvarea unor probleme practice | | | Teme acasa | | 10% |
| Rezolvarea unor probleme practice | | | Proiect | | 40% |
| 10.7 Standard minim de performanţă | | | | | | | |
| * **Minim 4.5 din 10 puncte; minim 50% din punctele de laborator/proiect; minim 50% din punctele obtinute pentru activitatea pe parcurs** | | | | | | | |
| **Data completării** | | | **Semnătura titularului de curs** | | **Semnătura titularului/ titularilor de aplicaţii** | |
| **Data avizării în departament** | | **Semnătura directorului de Departament** | | | | |
|  | | Prof. Dr. Ing. George DRAGOI | | | | |

**Annex no. 2**

**SUBJECT DESCRIPTION**

1. **Information about the program**

|  |  |
| --- | --- |
| * 1. Higher education institution | University POLITEHNICA of Bucharest |
| 1.2 Faculty | Faculty of Engineering in Foreign Languages |
| 1.3 Department | Department of Engineering in Foreign Languages |
| 1.4 Field of study | Computer and Information Technology |
| 1.5 Study cycle | Undergraduate (bachelor) |
| 1.6 Program / Qualification | Internet of Things Engineering (in English)/ Engineer |

1. **Data about the subject**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 2.1 Name of subject | | | Data Structures and Algorithms | | | | |
| 2.2 Course holder | | |  | | | | |
| 2.3 Seminar holder | | |  | | | | |
| 2.4 Laboratory/project holder | | |  | | | | |
| 2.5 Year of study | 1 | 2.6 Semester | 2 | 2.7 Evaluation type | E | 2.8 Subject type | DD/DO |

1. **Estimated time (hours per semester) of didactic activities**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **3.1 Number of hours per week** | 4 | course hours | | | 2 | seminar |  | laboratory | 2 |
| **3.2. Number of hours per semester** | 56 | course hours | | | 28 | seminar |  | laboratory | 28 |
| **3.3.Distribution of spend time:** | | | | | | | |  | **h** |
| Study of textbooks, bibliography and course notes | | | | | | | |  | 14 |
| Supplementary study in library, on electronic platforms, on the fieldwork | | | | | | | |  | 6 |
| Preparation of seminars/laboratories, home assignments, papers, portfolios, essays | | | | | | | |  | 16 |
| Tutoring | | | | | | | |  | 10 |
| Examinations | | | | | | | |  | 2 |
| Other activities | | | | | | | |  |  |
| **3.4 Total hours of individual study** |  | | **48** |
| **3.5 Total hours per semester[[7]](#footnote-7)** |  | | **104** |
| **3.6 Number of credits** |  | | **4** |

1. **Preconditions (where relevant)**

|  |  |
| --- | --- |
| 4.1 curriculum- related | * attendance and/or obtaining credit for the following courses: Introduction to Information Technology, Programming Languages |
| 4.2 competence - related |  |

1. **Facilities and equipment (where relevant)**

|  |  |
| --- | --- |
| 5.1 for the course | * Lecture room with project |
| 5.2 for the course seminar |  |
| 5.3 for the laboratory/project | * Room with computers and internet connection |

1. **Specific competences acquired**

|  |  |
| --- | --- |
| Professional competences | * Ability to perform computations, proofs and applications for solving tasks specific to engineering, based on knowledge from fundamental and engineering sciences and from new information and communication technologies. * Developing knowledge, digital technologies and software applications for implementing integrated, intelligent products and for managing conformity with quality and security standards in ITC. |
| Transversal competences | * Adaptation to new technologies, professional and personal development through continuous training using printed documentation, specialized software and electronic resources in Romanian and at least one foreign language |

1. **Course objectives (as resulting from the grid of specific competences)**

|  |  |
| --- | --- |
| 7.1 Subject general goal | * the design, development and implementation of the fundamental data structures and data structure-related algorithms in the C/C++ programming language |
| 7.2 Specific objectives | * implementation of the data structures studied during the course (as well as some simple extensions of these data structures) in the context of very concrete problems * learning C/C++ programing language |

1. **Content**

|  |  |  |
| --- | --- | --- |
| **8.1 Course** | **Teaching methods** | **Observations** |
| Introduction to C/C++ (Basics of C/C++, static arrays, matrices, multidimensional arrays, struct vs classes, template classes, recursive functions) | Lecture | 5 hours |
| Dynamic Memory Allocation in C/C++ | Lecture | 2 hours |
| Abstract Data Type vs Data Structure | Lecture | 1 hour |
| Stack | Lecture | 2 hours |
| Queue | Lecture | 2 hours |
| Linked Lists. Stacks and Queues as Linked Lists | Lecture | 4 hours |
| Hash Tables | Lecture | 2 hours |
| Graphs | Lecture | 4 hours |
| Binary Trees | Lecture | 2 hours |
| Binary Search Trees | Lecture | 4 hours |
| Heaps | Lecture | 2 hours |
| **Bibliography**  1. “C++ Programming Language”, Bjarne Stroustroup  2. “Thinking in C++”, by Bruce Eckel & Chuck Allison  3. “C++ Plus Data Structures”, by Nell Dale  4. M. A. Weiss, “Data Structures and Algorithm Analysis in C++”, 3rd edition, Addison-Wesley, 2007  5. T.H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein, "Introduction to Algorithms", 2nd edition, MIT Press, 2002 | | |
| **8.2 Seminar** | **Teaching methods** | **Observations** |
|  |  |  |
| **8.3 Laboratory** |  |  |
| Programming in C/C++ (Objectives: to run and compile C programs; to identify the structure of a C program; to use standard I/O operations; to define variables; to declare and implement functions; to make structures; to make the difference between C and C++; struct vs. classes in C++; template classes and function templates) | Demo, exercises | 4 hours |
| Sorting algorithms and recursive functions | Demo, exercises | 2 hours |
| Pointers in C/C++ | Demo, exercises | 2 hours |
| Stack (Basic operations, implementations and applications - Reversed Polish Notation) | Demo, exercises | 2 hours |
| Queue (Basic operations, implementations and applications) | Demo, exercises | 2 hours |
| Linked Lists (Various types of linked lists, implementations and applications. Using Linked Lists to implement stacks and queues) | Demo, exercises | 2 hours |
| Hash-tables (Basic operations and implementations. Hash Functions) | Demo, exercises | 2 hours |
| Graphs (Representations and searching algorithms – DFS vs. BFS. Applications: topological sort, shortest path algorithms, bipartite graphs) | Demo, exercises | 4 hours |
| Binary Trees | Demo, exercises | 2 hours |
| Binary Search Trees | Demo, exercises | 4 hours |
| Heaps and Heapsort | Demo, exercises | 2 hours |
| **Bibliography**  Same from the course | | |

1. **Subjects’ relevance to the epistemic community, professional associations and representative employers in fields significant for the program**

|  |
| --- |
| * **The proposed exercises are aligned to the questions from professional interviews** |

1. **Assessment**

|  |  |  |  |
| --- | --- | --- | --- |
| Activity | 10.1 Evaluation criteria | 10.2 Evaluation methods | 10.3 Weight in final grade |
| 10.4 Course | Concepts taught during the course | Written exam | 20% |
| Ability of applying the concepts taught during the course in exercises | Written exam | 30% |
| 10.5 Seminar |  |  |  |
|  |  |  |
| 10.6 Laboratory/Project | On-going activity | Practical exercises during the lab | 20% |
| Resolving practical problems | Homework | 30% |
| 10.7 Minimal standard of performance | | | |
| * Minimum 4.5 out of 10 points; minimum 25% from lab points | | | |

|  |  |  |
| --- | --- | --- |
| **Filling in date** | **Signature of the course holder** | **Signature of the applications holder(s)** |

|  |  |
| --- | --- |
| **Date of endorsement in the department** | **Signature of the Department Director** |
|  | Prof. Dr. Ing. George DRĂGOI |

**Anexa nr. 2**

**FIŞA DISCIPLINEI**

1. **Date despre program**

|  |  |
| --- | --- |
| * 1. Instituţia de învăţământ superior | Universitatea POLITEHNICA din Bucureşti |
| 1.2 Facultatea | Facultatea de Inginerie în Limbi Străine |
| 1.3 Departamentul | Departamentul de Inginerie în Limbi Străine |
| 1.4 Domeniul de studii | Calculatoare si tehnologia informatiei |
| 1.5 Ciclul de studii | Licență |
| 1.6 Programul de studii / Calificarea | Ingineria informatiei (în limba engleză)/Inginer |

1. **Date despre disciplină**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 2.1 Denumire disciplina | | | Structuri de date și algoritmi | | | | |
| 2.2 Titular activităţi de curs | | |  | | | | |
| 2.3 Titular activităţi de seminar | | |  | | | | |
| 2.4 Titular activităţi de laborator/lucrari | | |  | | | | |
| 2.5 Anul de studiu | 1 | 2.6 Semestrul | 2 | 2.7 Tipul de evaluare | E | 2.8 Regimul disciplinei | DD/DO |

1. **Timpul total estimat (ore pe semestru) al activităţilor didactice**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **3.1 Număr de ore pe săptămână** | 4 | din care ore curs | | | 2 | seminar |  | laborator | 2 |
| **3.2. Numar ore pe semestru** | 56 | din care ore curs | | | 28 | seminar |  | laborator | 28 |
| **3.3.Distribuţia fondului de timp:** | | | | | | | |  | **ore** |
| Studiul după manual, suport de curs, bibliografie şi notiţe | | | | | | | |  | 14 |
| Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate / pe teren | | | | | | | |  | 6 |
| Pregătire seminarii / laboratoare, teme, referate, portofolii şi eseuri | | | | | | | |  | 16 |
| Tutoriat | | | | | | | |  | 10 |
| Examinări | | | | | | | |  | 2 |
| Alte activităţi | | | | | | | |  | 0 |
| **3.4 Total ore studiu individual** |  | | **48** |
| **3.5 Total ore pe semestru[[8]](#footnote-8)** |  | | **104** |
| **3.6 Numărul de credite** |  | | **4** |

1. **Precondiţii (acolo unde este cazul)**

|  |  |
| --- | --- |
| * 1. de curriculum | * parcurgerea si/sau promovarea urmatoarelor discipline: Introducere in tehnologia informatiei, Limbaje de programare |
| 4.2 de competenţe |  |

1. **Condiţii (acolo unde este cazul)**

|  |  |
| --- | --- |
| 5.1 de desfăşurare a cursului | * Amfiteatru cu videoproiector |
| 5.2 de desfăşurare a seminarului |  |
| 5.3 de desfăşurare a laboratorului | * Sala cu calculatoare si conexiune la internet |

1. **Competenţele specifice acumulate**

|  |  |
| --- | --- |
| Competenţe profesionale | * Efectuarea de calcule, demonstrații și aplicații, pentru rezolvarea de sarcini specifice ingineriei, pe baza cunoștințelor din științele fundamentale, inginerești și a noilor tehnologii informaționale și de comunicații. * Dezvoltarea de cunoștințe, tehnologii digitale și aplicații software pentru realizarea de produse inteligente integrate și gestionarea conformitații cu standardele de calitate și securitate în TIC. |
| Competenţe transversale | * Adaptarea la noile tehnologii, dezvoltarea profesională şi personală, prin formare continuă folosind surse de documentare tipărite, software specializat şi resurse electronice în limba română şi, cel puţin, într-o limbă de circulaţieinternaţională. |

1. **Obiectivele disciplinei (reieşind din grila competenţelor specifice acumulate)**

|  |  |
| --- | --- |
| 7.1 Obiectivul general al disciplinei | * proiectarea, dezvoltarea şi implementarea a câtorva structuri de date fundamentale şi a unor algoritmi bazaţi pe structuri de date, în C/C++ |
| 7.2 Obiectivele specifice | * implementareastructurilor de date studiate la curs (precum şi nişte extensii simple ale acestora), în contextul unor probleme foarte concrete * invatarea limbajului C/C++ |

1. **Conţinuturi**

|  |  |  |
| --- | --- | --- |
| **8.1 Curs** | **Metode de predare** | **Observaţii** |
| Introducere in C/C++ (Notiuni fundamentale de C/C++, tablouri statice, matrici, tablouri multidimensionale, struct vs clase, clase sablon, functii recursive) | Prelegere | 5 ore |
| Alocarea dinamica a memoriei in C/C++ | Prelegere | 2 ore |
| Tipul abstract de date vs. structure de date | Prelegere | 1 ora |
| Stiva | Prelegere | 2 ore |
| Coada | Prelegere | 2 ore |
| Listele inlantuite. Implementarea tipurilor de date Stivă şi Coadă folosind liste înlănţuite | Prelegere | 4 ore |
| Tabele de dispersie | Prelegere | 2 ore |
| Grafuri | Prelegere | 4 ore |
| Arbori binari | Prelegere | 2 ore |
| Arbori binari de cautare | Prelegere | 4 ore |
| Heap-uri | Prelegere | 2 ore |
| **Bibliografie**  1. “C++ Programming Language”, Bjarne Stroustroup  2. “Thinking in C++”, by Bruce Eckel & Chuck Allison  3. “C++ Plus Data Structures”, by Nell Dale  4. M. A. Weiss, “Data Structures and Algorithm Analysis in C++”, 3rd edition, Addison-Wesley, 2007  5. T.H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein, "Introduction to Algorithms", 2nd edition, MIT Press, 2002 | | |
| **8.2 Seminar** | **Metode de predare** | **Observaţii** |
|  |  |  |
| **8.3 Laborator** |  |  |
| Programare in C/C++ (Obiective: compilarea si executia programelor in C; identificarea structurii unui program in C; utilizarea operatiilor de I/O standard; definirea variabilelor; declararea si implementarea functiilor; constructia structurilor; diferentierea intre C si C++; struct vs. clase; clase sablon si functii sablon) | Demo, exercitii | 4 ore |
| Algoritmi de sortare si functii recursive | Demo, exercitii | 2 ore |
| Pointeri in C/C++ | Demo, exercitii | 2 ore |
| Stiva (Operatii de baza, implementari si aplicatii – forma poloneza inversata) | Demo, exercitii | 2 ore |
| Coada (Operatii de baza, implementari si aplicatii) | Demo, exercitii | 2 ore |
| Listeleinlantuite (tipuri diverse de liste inlantuite, implementari si aplicatii. Utilizarea listelor inlantuite pentru a implementa stive si cozi) | Demo, exercitii | 2 ore |
| Tabele de dispersie (Operatii de baza si implementari. Functii hash) | Demo, exercitii | 2 ore |
| Grafuri (Reprezentari si algoritmi de cautare – DFS vs. BFS. Aplicatii: sortare topologica, algoritmi pentru a afla drumul minim, grafuri bipartite) | Demo, exercitii | 4 ore |
| Arboribinari | Demo, exercitii | 2 ore |
| Arbori binary de cautare | Demo, exercitii | 4 ore |
| Heap-uri si sortarea folosind heap-uri | Demo, exercitii | 2 ore |
| **Bibliografie**  Cea mentionata la curs | | |

1. **Colaborarea conţinuturilor disciplinei cu aşteptările reprezentanţilor comunităţii epistemice, asociaţiilor profesionale şi angajatori reprezentativi din domeniul aferent programului**

|  |
| --- |
| * Exercitiile propuse sunt aliniate intrebarilor din interviurile profesionale. |

1. **Evaluare**

|  |  |  |  |
| --- | --- | --- | --- |
| Tip activitate | 10.1Criterii de evaluare | 10.2 Metode de evaluare | 10.3 Pondere din nota finală |
| 10.4 Curs | Conceptele predate la curs | Examen scris | 20% |
| Capacitatea de aplicare a conceptelor predate la curs prin rezolvarea exercitiilor | Examen scris | 30% |
| 10.5 Seminar |  |  |  |
|  |  |  |
| 10.6 Laborator/lucrari | Activitatea pe parcurs | Exercitii practice in timpul laboratorului | 20% |
| Rezolvarea unor probleme practice | Teme acasa | 30% |
| 10.7 Standard minim de performanţă | | | |
| * Minim 4.5 din 10 puncte; minim 50% din punctele de laborator | | | |

|  |  |  |
| --- | --- | --- |
| **Data completării** | **Semnătura titularului de curs** | **Semnătura titularului/ titularilor de aplicaţii** |

|  |  |
| --- | --- |
| **Data avizării în departament** | **Semnătura directorului de Departament** |
|  | Prof. Dr. Ing. George DRĂGOI |

1. Numărul total de ore nu trebuie să depăşească valoarea (Număr credite) x 27 ore [↑](#footnote-ref-1)
2. Numărul total de ore nu trebuie să depăşească valoarea (Număr credite) x 27 ore [↑](#footnote-ref-2)
3. Numărul total de ore nu trebuie să depăşească valoarea (Număr credite) x 27 ore [↑](#footnote-ref-3)
4. Numărul total de ore nu trebuie să depăşească valoarea (Număr credite) x 27 ore [↑](#footnote-ref-4)
5. Numărul total de ore nu trebuie să depăşească valoarea (Număr credite) x 27 ore [↑](#footnote-ref-5)
6. Numărul total de ore nu trebuie să depăşească valoarea (Număr credite) x 27 ore [↑](#footnote-ref-6)
7. Numărul total de ore nu trebuie să depăşească valoarea (Număr credite) x 27 ore [↑](#footnote-ref-7)
8. Numărul total de ore nu trebuie să depăşească valoarea (Număr credite) x 27 ore [↑](#footnote-ref-8)