UNIVERSIDADE DA CORUÑA

Teaching Guide			
Identifying Data			2023/24
Subject (*)	Programming II	Code	614G01006
Study programme	Grao en Enxeñaría Informática		
Descriptors			

Cycle	Period	Year	Type	Credits
Graduate	2nd four-month period	First	Basic training	6

Language	SpanishGalicianEnglish		
Teaching method	Face-to-face		
Prerequisites			
Department	Ciencias da Computación e Tecnoloxías da InformaciónComputación		
Coordinador	Vilares Ferro, Jesus	E-mail	jesus.vilares@udc.es
Lecturers	Alonso Pardo, Miguel angel Cabado Lousa, Bruno Cabrero Canosa, Mariano Javier Guerreiro Santalla, Sara Guijarro Berdiñas, Berta M. Hernandez Pereira, Elena Maria Magaz Romero, Samuel Morán Fernández, Laura Naya Varela , Martín Paz Ruza, Jorge Pérez Sánchez, Beatriz Perez-Campoamor Manzaneque, Antonio Rodríguez Arias, Alejandro Romero Montero, Alejandro Sanchez Maroño, Noelia Suárez Marcote, Samuel Vilares Calvo, David Vilares Ferro, Jesus Zamaraeva , Olga	E-mail	miguel.alonso@udc.es bruno.cabado@udc.es mariano.cabrero@udc.es sara.guerreiro@udc.es berta.guijarro@udc.es elena.hernandez@udc.es s.magazr@udc.es laura.moranf@udc.es martin.naya@udc.es j.ruza@udc.es beatriz.perezs@udc.es a.perez-campoamor@udc.es alejanro.rodriguez.arias@udc.es alejandro.romero.montero@udc.es noelia.sanchez@udc.es s.suarez.marcote@udc.es david.vilares@udc.es jesus.vilares@udc.es olga.zamaraeva@udc.es
Web	campusvirtual.udc.gal		
General description	The subject focuses on programming with dynamic and complex data structures, dealed from the perspective of abstract data types. It belongs to the block of subjects of Languages and Programming of the Basic Training Module of the degree. It has a strong relationship with the subjects of Programming I, Databases, Algorithms and Software Design. It is also somehow related with mathematical subjects, especially with the Discrete Mathematics subject. Regarding the professional profile, many areas of computing require the ability to work with the data structures studied in this subject that will allow students to improve their skills as programmers.		

Study programme competences	
Code	Study programme competences
A3	Capacidade para comprender e dominar os conceptos básicos de matemática discreta, lóxica, algorítmica e complexidade computacional e a súa aplicación para a resolución de problemas propios da enxeñaría.
A4	Coñecementos básicos sobre o uso e a programación dos ordenadores, sistemas operativos, bases de datos e programas informáticos con aplicación na enxeñaría.
B1	Capacidade de resolución de problemas
B3	Capacidade de análise e síntese

C3	Utilizar as ferramentas básicas das tecnoloxías da información e as comunicacións (TIC) necesarias para o exercicio da súa profesión e para a aprendizaxe ao longo da súa vida.
C6	Valorar criticamente o coñecemento, a tecnoloxía e a información dispoñible para resolver os problemas cos que deben enfrontarse.

Learning outcomes			
Learning outcomes	Study programme competences		
Understanding the mechanisms of dynamic memory management.	A4	B1	C6
Understanding the mechanisms of abstraction in the design of data structures.	A4	B1 B3	C3 C6
Building specifications, designing the abstract type from them, using appropriate data structures.	A3 A4	B1 B3	C3 C6
Using appropriate data structures and program algorithms to solve real problems.	A3 A4	B1 B3	C3 C6
Assuming the need for a good specification and a good design as steps prior to coding.	A4	B3	C6
Internalizing good programming practices.	A4	B3	

Contents	
Topic	Sub-topic
Dynamic Memory Management	Program memory organization. Definition of pointer variables. Dynamic memory allocation and deallocation. Pointer assignment and comparison.
Introduction to Abstract Data Types (ADT)	Abstraction in programming: Concept, Evolution through computer programming history, ADT and Object Oriented Programming. Modularity in programming. Abstract Data Types: Definition and concept, Differences between datatype, data structure and ADT, ADT construction, Advantages of data abstraction.
Lists	Informal specification of List ADT. Implementations of List ADT. Ordered list ADT: specification and implementations. Multilists and multiordered lists: concept, representations and usage.
Stacks	Informal specification of Stack ADT. Implementations of Stack ADT. Applications on computer science.
Queues	Informal specification of Queue ADT. Implementations of Queue ADT. Queue variants. Priority queues. Applications on computer science.
Trees	Tree definition and terminology. Binary Tree ADT: Informal specification, Implementation. Binary Tree traversals.
Binary Search Trees	Binary Search Trees. AVL Trees.

Planning				
Methodologies / tests	Competencies	Ordinary class hours	Student?s personal work hours	Total hours
Guest lecture / keynote speech	A3 A4 B1 B3	30	30	60
Problem solving	A3 B1 B3 C6	10	14	24

Laboratory practice	A4 B1 B3 C3 C6	20	26	46
Objective test	A3 A4 B1 B3	3	15	18
Personalized attention		2	0	2
(*)The information in the planning table is for guidance only and does not take into account the heterogeneity of the students.				

Methodologies	
Methodologies	**Description**
Guest lecture / keynote speech	Used for face-to-face theory lectures. The teacher will make a brief description of the topics and objectives , in order to provide students with an overview of the subject. In addition, they will establish relationships with other concepts previously acquired to build a timeline, and set out the recommended bibliography. They will then develop the theoretical contents.
Problem solving	In the problem solving lectures, practical cases directly related with the theoretical concepts will be presented in order to reinforce those concepts. Initially, they will be resolved by the teacher to guide students. As the development of theoretical content advances, students will solve problems organized into working groups. When the examples used in problem solving lectures or theoretical explanations involve coding or pseudocode, they will be developed showing the successive steps of top-down design. The reason is twofold: (a) to get the student used to employ this method; and (b) to avoid getting lost in the details of the particular syntax and features of the language, instead of paying attention to the understanding and design of the solution. Additional exercises will be assigned as extra-classroom activities. The student must solve them and comment/correct them with the teacher during group and/or individual tutoring . The purpose is to encourage the participation of students and promote, as far as possible, open dialogue and evaluation of solutions.
Laboratory practice	Practical lectures will consist in the development of continuous and incremental practicals that require the students to program data structures in a high-level language. Regular delivery milestones will be proposed as follow-ups to encourage continued study. Practical project instructions will detail the nature of the problem to solve and its specifications, which must be strictly observed. These instructions will be provided sufficiently in advance for the students to read it carefully and analyze it in depth. Subsequently, the role of the teacher will be to oversee the practical sessions, solving doubts and correcting misunderstandings, bad programming habits and syntax errors, etc.
Objective test	Summative evaluation of the student through a final exam at the end of the fourth-month period, so that the student can demonstrate that he/she has acquired the required skills on abstraction, design, implementation and use of ADTs, and that he/she is sufficiently trained to use the precise skills to solve practical cases involving the application of such structures.

Personalized attention	
Methodologies	**Description**

Problem solving Laboratory practice Objective test	The development of theory lectures, problem-solving sessions and practical sessions will be carried out taking into account the progress of the students. The general progress of the class will be combined with specific attention to give additional support or expand knowledge. Laboratory practices will be carried out, in part, as autonomous work. For its correct development, periodic monitoring will be necessary to allow students to clarify errors of concept as soon as possible and to ensure the quality of the work. In both cases, Moodle will be used to make available to the students "thematic forums" that resolve the general doubts detected related to specific activities such as the practices or problems proposed. Individual tutoring should not be used to extend the contents with new concepts, but to clarify the concepts already discussed in class. The teacher will use them as an interaction that allows him/her to draw conclusions about the degree of assimilation of the subject by the students. Attention is maintained in the official tutoring hours through the following channels: - Email: Of use to make short answer queries. - Teams: virtual meetings preferably upon request via email.

Assessment			
Methodologies	**Competencies**	**Description**	**Qualification**
Problem solving	A3 B1 B3 C6	The results, form and conditions of completion of various assessable works that will be detailed during the course and that will be resolved during the SMALL GROUP TUTORIALS will be assessed. The result of the activity, as well as the discussion and active participation in class, will be valued in the final grade. This grades of these activities will only be added to the remaining grades once the course is passed.	10
Laboratory practice	A4 B1 B3 C3 C6	The practical works are mandatory according to the conditions stablished in each problem assignment. There will be a periodic follow-up of the development of these practicals throughout the course that will influence their grades. Students must present all the practical assignments and obtain a global minimum of 4.5 out of 10 points to pass the subject. The work submitted must be the student's original work. According to Article 11, Section 4 b) of the UDC Student Disciplinary Regulations (&quot;Reglamento disciplinar del estudiantado de la Universidad de A Coruña&quot;). the submission of non-original work or work with duplicate parts (either by copying among classmates, obtaining from other sources, etc.) will result in a grade of FAIL with 0 points in that full ANNUAL CALL for both the group that employed copied material and the group that provided it.	40
Objective test	A3 A4 B1 B3	Mandatory realization. It implies a global treatment of the contents covered throughout the course. It will be eminently practical so that students can demonstrate that they have acquired the necessary knowledge of abstraction, design, implementation and use of ADTs and that they have trained sufficiently in the skills required by the course. A minimum grade of 4,5 out of 10 is required to pass the course.	60

Others			

Assessment comments

On attending
the practical classesRepeated
non-attendance could carry a penalty in the grade according to the specific
conditions that will be detailed at the beginning of the course.
On shared
responsibility for group workIn the activities carried out in groups, such as practicals,
all members of the group will be jointly liable for the work carried out and
delivered, as well as the consequences arising from non-compliance with the
rules of authorship of the work.
About the final
note of the minutesIn the event
that the conditions for the joint calculation of the grade between the final
test and the practicals are not met (that is, if either of the two is less than
4.5 out of 10), the minutes will include the minimum mark between 4.5 and the
one resulting from the joint calculation.
Second Chance EvaluationThe mark of the exam ("Objective test") is NOT kept between opportunities.
The marks of the "Laboratory practices" as
well as the block of "Problem Solving" will be kept for the second
opportunity. Only lab practicals qualified with FAIL ("Suspenso") or NOT SUBMITTED ("No Presentado") in the first
opportunity may be delivered in the second, always according to the statement
that is proposed for it.
In line with the UDC evaluation regulations, those grades derived from other activities developed throughout the course as
part of the continuous evaluation (introductory practicals, follow-ups of lab practicals, tests in
Small Tutoring Groups, etc.) will not be recoverable for the second opportunity.
Regarding the evaluation criteria, the second
opportunity will keep the same as the first one.
Part-time
enrollmentIn the case of students with part-time enrollment,
the obligation to attend practical lectures is eliminated. However, they will
have to deliver the assessable activities under the specific conditions and
deadlines that will be established during the course. It is the responsibility
of these students to inform the teacher of their circumstances.
Advanced
Opportunity in DecemberThe evaluation of this advanced opportunity will be based
exclusively on a written test.

```
@font-face
 {font-family:"Courier New";
 panose-1:2 7 3 9 2 2 5 2 4 4;
 mso-font-charset:0;
 mso-generic-font-family:auto;
 mso-font-pitch:variable;
 mso-font-signature:-536859905 -1073711037 9 0 511 0;}@font-face
 {font-family:Wingdings;
 panose-1:5 0 0 0 0 0 0 0 0 0;
 mso-font-charset:2;
 mso-generic-font-family:auto;
 mso-font-pitch:variable;
 mso-font-signature:0 268435456 0 0 -2147483648 0;}@font-face
 {font-family:"?? ??";
 mso-font-charset:78;
```

```
mso-generic-font-family:auto;
mso-font-pitch:variable;
mso-font-signature:-536870145 1791491579 18 0 131231 0;}@font-face
{font-family:Verdana;
panose-1:2 11 6 4 3 5 4 4 2 4;
mso-font-charset:0;
mso-generic-font-family:auto;
mso-font-pitch:variable;
mso-font-signature:-1593833729 1073750107 16 0 415 0;}@font-face
{font-family:"Cambria Math";
panose-1:2 4 5 3 5 4 6 3 2 4;
mso-font-charset:0;
mso-generic-font-family:auto;
mso-font-pitch:variable;
mso-font-signature:-536870145 1107305727 0 0 415 0;}@font-face
{font-family:Cambria;
panose-1:2 4 5 3 5 4 6 3 2 4;
mso-font-charset:0;
mso-generic-font-family:auto;
mso-font-pitch:variable;
mso-font-signature:-536870145 1073743103 0 0 415 0;}p.MsoNormal, li.MsoNormal, div.MsoNormal
{mso-style-unhide:no;
mso-style-qformat:yes;
mso-style-parent:"";
margin-top:0cm;
margin-right:0cm;
margin-bottom:10.0pt;
margin-left:0cm;
mso-pagination:widow-orphan;
font-size:12.0pt;
font-family:Cambria;
mso-ascii-font-family:Cambria;
mso-ascii-theme-font:minor-latin;
mso-fareast-font-family:"?? ??";
mso-fareast-theme-font:minor-fareast;
mso-hansi-font-family:Cambria;
mso-hansi-theme-font:minor-latin;
mso-bidi-font-family:"Times New Roman";
mso-bidi-theme-font:minor-bidi;
mso-fareast-language:JA;}p.MsoListParagraph, li.MsoListParagraph, div.MsoListParagraph
{mso-style-priority:34;
mso-style-unhide:no;
mso-style-qformat:yes;
margin-top:0cm;
margin-right:0cm;
margin-bottom:10.0pt;
margin-left:36.0pt;
mso-add-space:auto;
mso-pagination:widow-orphan;
font-size:12.0pt;
```

```
font-family:Cambria;
mso-ascii-font-family:Cambria;
mso-ascii-theme-font:minor-latin;
mso-fareast-font-family:"?? ??";
mso-fareast-theme-font:minor-fareast;
mso-hansi-font-family:Cambria;
mso-hansi-theme-font:minor-latin;
mso-bidi-font-family:"Times New Roman";
mso-bidi-theme-font:minor-bidi;
mso-fareast-language:JA;}p.MsoListParagraphCxSpFirst, li.MsoListParagraphCxSpFirst, div.MsoListParagraphCxSpFirst
{mso-style-priority:34;
mso-style-unhide:no;
mso-style-qformat:yes;
mso-style-type:export-only;
margin-top:0cm;
margin-right:0cm;
margin-bottom:0cm;
margin-left:36.0pt;
margin-bottom:.0001pt;
mso-add-space:auto;
mso-pagination:widow-orphan;
font-size:12.0pt;
font-family:Cambria;
mso-ascii-font-family:Cambria;
mso-ascii-theme-font:minor-latin;
mso-fareast-font-family:"?? ??";
mso-fareast-theme-font:minor-fareast;
mso-hansi-font-family:Cambria;
mso-hansi-theme-font:minor-latin;
mso-bidi-font-family:"Times New Roman";
mso-bidi-theme-font:minor-bidi;
mso-fareast-language:JA;}p.MsoListParagraphCxSpMiddle, li.MsoListParagraphCxSpMiddle, div.MsoListParagraphCxSpMiddle
{mso-style-priority:34;
mso-style-unhide:no;
mso-style-qformat:yes;
mso-style-type:export-only;
margin-top:0cm;
margin-right:0cm;
margin-bottom:0cm;
margin-left:36.0pt;
margin-bottom:.0001pt;
mso-add-space:auto;
mso-pagination:widow-orphan;
font-size:12.0pt;
font-family:Cambria;
mso-ascii-font-family:Cambria;
mso-ascii-theme-font:minor-latin;
mso-fareast-font-family:"?? ??";
mso-fareast-theme-font:minor-fareast;
mso-hansi-font-family:Cambria;
```

```
mso-hansi-theme-font:minor-latin;
mso-bidi-font-family:"Times New Roman";
mso-bidi-theme-font:minor-bidi;
mso-fareast-language:JA;}p.MsoListParagraphCxSpLast, li.MsoListParagraphCxSpLast, div.MsoListParagraphCxSpLast
{mso-style-priority:34;
mso-style-unhide:no;
mso-style-qformat:yes;
mso-style-type:export-only;
margin-top:0cm;
margin-right:0cm;
margin-bottom:10.0pt;
margin-left:36.0pt;
mso-add-space:auto;
mso-pagination:widow-orphan;
font-size:12.0pt;
font-family:Cambria;
mso-ascii-font-family:Cambria;
mso-ascii-theme-font:minor-latin;
mso-fareast-font-family:"?? ??";
mso-fareast-theme-font:minor-fareast;
mso-hansi-font-family:Cambria;
mso-hansi-theme-font:minor-latin;
mso-bidi-font-family:"Times New Roman";
mso-bidi-theme-font:minor-bidi;
mso-fareast-language:JA;}.MsoChpDefault
{mso-style-type:export-only;
mso-default-props:yes;
font-family:Cambria;
mso-ascii-font-family:Cambria;
mso-ascii-theme-font:minor-latin;
mso-fareast-font-family:"?? ??";
mso-fareast-theme-font:minor-fareast;
mso-hansi-font-family:Cambria;
mso-hansi-theme-font:minor-latin;
mso-bidi-font-family:"Times New Roman";
mso-bidi-theme-font:minor-bidi;
mso-fareast-language:JA;}.MsoPapDefault
{mso-style-type:export-only;
margin-bottom:10.0pt;}div.WordSection1
{page:WordSection1;}ol
{margin-bottom:0cm;}ul
{margin-bottom:0cm;}
```

Sources of information	
Basic	- Narasimha Karumanchi (2017). Data Structures and Algorithms Made Easy, 5th Edition. CareerMonk Publications - Ignacio Zahonero y Luis Joyanes Aguilar (2004). Algoritmos y estructuras de datos: Una perspectiva en C. McGraw-Hill - Kyle Loudon (1999). Mastering Algorithms with C. O'Reilly Media <br />
Complementary	- Reema Thareja (2014). Data Structures Using C - Second Edition. Oxford University Press - Aaron M. Tenenbaum, Yedidyah Langsam &amp; Moshe J. Augenstein (1989). Data Structures Using C. Prentice Hall - Richard F. Gilberg &amp; Behrouz A. Forouzan (2005). Data Structures: A Pseudocode Approach with C (2nd Ed). Cengage Learning - Ignacio Zahonero y Luis Joyanes Aguilar (2005). Programación en C. Metodología, Algoritmos y Estructura de Datos, 2º Edición. McGraw-Hill - Luis Joyanes Aguilar, Andrés Castillo Sanz, Lucas Sánchez García e Ignacio Zahonero Martínez (2002). Programación en C. Libro de problemas. McGraw-Hill - Ignacio Zahonero, Lucas García Sánchez, Luis Joyanes Aguilar y Matilde Fernández Azuela (2005). Estructuras de datos en C (Serie Schaum). McGraw-Hill Páxina oficial da contorna de desenvolvemento CLion: https://www.jetbrains.com/clion/

Recommendations
Subjects that it is recommended to have taken before
Programming I/614G01001 Computer Science Preliminaries/614G01002 Discrete Mathematics/614G01004
Subjects that are recommended to be taken simultaneously
Subjects that continue the syllabus
Algorithms/614G01011 Databases/614G01013 Programming Paradigms/614G01014 Software Design/614G01015
Other comments
DISCREPANCIES BETWEEN LANGUAGESIn case of discrepancies in the interpretation of the different translations of the Teaching Guide of this course, the Galician Teaching Guide will be taken as a reference.GENDER PERSPECTIVEAs stated in the different regulations applicable to university teaching, this course incorporates the gender perspective. The intervention of male and female students in class will be encouraged, and work will be done to identify and modify sexist prejudices and attitudes, and the environment will be influenced to modify them and promote values of respect and equality. Situations of gender discrimination should be detected and actions and measures to correct them will be proposed.

(*)The teaching guide is the document in which the URV publishes the information about all its courses. It is a public document and cannot be modified. Only in exceptional cases can it be revised by the competent agent or duly revised so that it is in line with current legislation.