



@prog2ua

Unit 0: Overview

Programming 2

Degree in Computer Engineering
University of Alicante
2024-2025



WHO ARE WE?



CSE STUDENTS!



**WHAT ARE OUR
EXPECTATIONS?**



**CODING GAMES
AND HACKING!**



**WHAT ARE WE
ACTUALLY DOING?**



cout << "HELLO WORLD!";



- Core subject in the Computer Science Degree
- 6 ECTS credits
- Main (official) policies for the course can be found at:
<https://cvnet.cpd.ua.es/Guia-Docente/?wlengua=en&wcodasi=34008&scaca=2024-25>
- **Course materials can be found in Moodle via UACloud**
- News and tutoring through UACloud and Moodle
- Follow us on X: @prog2ua

- Coordinator:
 - David Tomás
- Theoretical sessions teachers:
 - Paco Castellanos, Eliseo Fuentes, Mariano López, Maribel Jiménez (I2ADE), and David Tomás (VAL, ARA, I2ADE)
- Lab sessions teachers:
 - Gonzalo Alcalá, Matías Díaz, Felicidad García, Maribel Jiménez, Fernando Llopis, Paco Moreno, and Antonio Pertusa

- Face to face:
 - In the office or by video conference
 - Office hours available at <http://www.dlsi.ua.es>
 - Necessary to book an appointment via UACloud
- Online:
 - Using UACloud (questions are answered as soon as possible)
 - Do not make enquiries by e-mail

"C++ makes it harder to shoot yourself, but when you do it blows your whole leg off"

Bjarne Stroustrup, C++ creator

- Unit 1. Introduction
- Unit 2. The `string` class
- Unit 3. Files
- Unit 4. Dynamic memory
- Unit 5. Introduction to object oriented programming

Aims

- To analyse the requirements of a problem
- To design and implement mid-sized programs
- To develop abstraction and generalisation skills
- To properly organise a program using functions and classes
- To identify the most efficient solutions
- To implement software using an adequate and understandable programming style
- To develop criticism in the algorithm verification process
- To use basic programming tools
- To learn basic notions of object oriented programming

Evaluation (1/3)

- Theory (50% of the final grade):
 - Multiple-choice test in the theory classroom
 - Theoretical concepts and short exercises
 - **Date: May 30, 2025**
- Practical work (50% of the final grade):
 - Three assignments: $p1$, $p2$, and $p3$
 - Graded as *PASS* or *FAIL* using an auto-grader (*autocorrector*)
 - Passing them grants the right to take the practical exam
 - **Practical grade = $0.3 * ex.p1 + 0.3 * ex.p2 + 0.4 * ex.p3$**

Evaluation (2/3)

- To average between theory and practical work, a minimum score of 4 is required in both parts
- The final grade must be equal to or greater than 5 to pass
- Algorithm to calculate the final grade:

```
practicalGrade = 0.3*ex.p1 + 0.3*ex.p2 + 0.4*ex.p3;  
  
if (theoryGrade >= 4 && practicalGrade >= 4)  
    finalGrade = 0.5*theoryGrade + 0.5*practicalGrade;  
else  
    finalGrade = kFAIL;
```

Evaluation (3/3)

- In case of failing in June:
 - The practical grade is preserved for July if it is greater than or equal to 4
 - The theory grade is not preserved for July
 - If a minimum score of 4 is not obtained in practical work in June, it will not be possible to pass during that session (you cannot average), and therefore, taking the theory exam in June does not make sense (since the grade is not preserved for July)
- July session:
 - Theory: exam on July 7, 2025
 - Practical work: you must achieve *PASS* in all three assignments with new tests and take an exam covering the content of all three

Lab sessions guidelines

- Your lab group can be consulted at UACloud
- It is not possible to change your assigned group (if justified reasons such as work or family issues exist, you may request group change at the secretariat of EPS)
- Lab sessions beginning: Monday, January 27
- **Assignments are done individually**
- **Attendance is mandatory: maximum 3 unjustified absences**
- It is necessary to strictly follow the instructions in the assessments (especially those regarding the output format of your program or submission instructions)

- The only way to learn to program is by programming
- Assessments are designed so that students learn in a practical way during the semester
- A student who copies (or who gets the assignment done by someone else) rarely learns and passes the course

- Regulation for student evaluation at UA, Article 14.1:
“Students must respect the rules about the authenticity of the presented work and its privacy.”
- Copying an assignment from another student, in whole or in part, violates the Article 14.1 (authenticity)
- Sharing an assignment (so other students could copy it or get inspiration), also violates Article 14.1 (privacy)

- Regulation for students evaluation at UA, Article 14.4:
“In any case, fraudulent actions in an evaluation procedure will result in a failing grade, with a numerical score of zero in that test, without prejudice to the disciplinary procedure that could be initiated against the student and, if appropriate, the sanction that could be applied in accordance with the present legislation”
- Students involved in cheating will get 0 in the corresponding assignment or exam and a report will be sent to EPS for subsequent disciplinary actions

Using AI in Programming 2

- We know you use **ChatGPT** (or similar AI tools) to help you develop your code
- It is a tool that will be essential in your career as programmers, and it is okay to use it, but...
- **What is allowed with AI in Programming 2?**
 - While programming
 - Obtain explanations about the use of functions
 - Document your code
 - Review code style
 - After finishing your code
 - Create tests
 - Debug/optimize
- **What NOT to do with AI in Programming 2?**
 - Ask it to write your code from scratch by simply providing the instructions...
 - ... because you will not learn...
 - ... **and during the exam, you are not allowed to use AI**

Temporal Planning

Monday	Wednesday	Thursday	Friday	Submission
27/01 T0	29/01 T0	30/01 T0	31/01 T0	-
03/02 T1 (1)	05/02 T1 (1)	06/02 T1 (1)	07/02 T1 (1)	-
10/02 T1 (2)	12/02 T1 (2)	13/02 T1 (2)	14/02 T1 (2)	-
17/02 T1 (3)	19/02 T1 (3)	20/02 T1 (3)	21/02 T1 (3)	<i>p1</i>
24/02 T2	26/02 T2	27/02 T2	28/02 T2	-
03/03 T3 (1)	05/03 T3 (1)	06/03 T3 (1)	07/03 T3 (1)	<i>ex1</i>
10/03 T3 (2)	12/03 T3 (2)	13/03 T3 (2)	14/03 T3 (2)	-
17/03 T4	19/03 -	20/03 T4	21/03 T4	-
24/03 T5 (1)	26/03 T4	27/03 T5 (1)	28/03 T5 (1)	<i>p2</i>
31/03 T5 (2)	02/04 T5 (1)	03/04 T5 (2)	04/04 T5 (2)	-
07/04 T5 (3)	09/04 T5 (2)	10/04 T5 (3)	11/04 T5 (3)	<i>ex2</i>
14/04 Exer.	16/04 T5 (3)	17/04 -	18/04 -	-
21/04 -	23/04 -	24/04 -	25/04 -	-
28/04 -	30/04 Exer.	01/05 -	02/05 -	-
05/05 T5 (4)	07/05 T5 (4)	08/05 T5 (4)	09/05 T5 (4)	<i>p3</i>
12/05 T5 (5)	14/05 T5 (5)	15/05 T5 (5)	16/05 T5 (5)	-
19/05 Exer.	21/05 Exer.	22/05 Exer.	23/05 Exer.	<i>ex3</i>

In order to pass Programming 2...

- You must practice a lot
- You need to do the theoretical exercises and the assignments
- You cannot pass by starting to study a week before the exam
- Raise your doubts in both theoretical and lab practice sessions
- If you get lost, do in-person or online tutoring