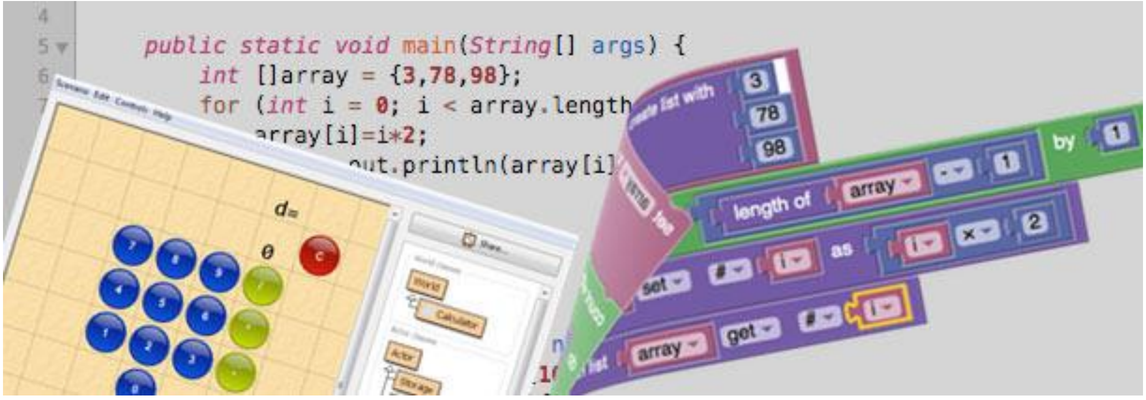
INTRODUCTION TO PROGRAMMING WITH JAVA



```
public static void main(String[] args) {
    int []array = {3,78,98};
    for (int i = 0; i < array.length
        array[i]=i*2;
        out.println(array[i]
```

Part III:
Fundamental Data Structures and Algorithms

LENGTH:	5 weeks
EFFORT:	5 - 7 hours per week
SUBJECT:	Computer Science
LEVEL:	Introductory
LANGUAGE:	English
VIDEO TRANSCRIPTS:	English, Español

SYLLABUS

uc3m | Universidad Carlos III de Madrid

INTRODUCTION

Introduction to Programming with Java is an introductory course to learn programming in an easy and interactive way. This course is designed taking into account the subset and recommendations of the College Board in order to prepare learners for the Advanced Placement (AP) Computer Science A exam. The course is divided in three parts of five weeks each. This distribution is intended to make it easier for learners to work and practice at their own pace.

Part 3: Fundamental Data Structures and Algorithms takes over from the second part, and aims to teach some of the most popular data structures and algorithms applied on them. Linear data structures, such as lists, stacks, queues and deques (double ended queues) are explained in the first weeks. After that, non-linear data structures, particularly trees, are introduced, with special emphasis on binary search trees and heaps. Algorithms for inserting, deleting and searching are explored on all these data structures, allocating a specific week for sorting algorithms which allow working with data structures in an efficient way, but that have additional complexity.

Learner's experience is enhanced through a wide range of videos and audiovisual materials. Interactive exercises accompanying videos give immediate feedback on common pitfalls and misconceptions when programming in Java. Code is presented using Codeboard, a web-based integrated development environment (IDE) that was also used in the first and second parts of this course. Students can use any IDE to follow the course. Although not mandatory, teachers recommend Eclipse IDE, which is a powerful and versatile environment that is very popular among software developers.

It is important that you know that some activities and exercises in this course may not be compatible with mobile devices (such as those running on iOS or Android). We recommend that you use a computer to ensure the best learning experience.

If this is your first course on edX, do not hesitate to enroll in the Demo course to get to know the courseware: https://www.edx.org/course/demox-edx-demox-1.

In addition, we recommend you to follow us in the first and second parts of this introductory programming course, which are also available on edX as self-paced courses: ***Introduction to Programming with Java 1: Starting to Code with Java*** and ***Introduction to Programming with Java 2: Writing Good Code*** https://www.edx.org/xseries/introduction-programming-java.

OBJECTIVES

After finishing ***Introduction to Programming with Java 3: Fundamental Data Structures and Algorithms***, the learner should:

- Arrange data on arrays and linked lists using primitive data types and classes.
- Develop and use linear data structures, such as stacks and queues, implemented with linked lists or arrays.
- Develop and use non-linear data structures, such as trees, including binary search trees and heaps, implemented with linked lists or arrays.
- Implement algorithms for the efficient searching and sorting of data.

COURSE STAFF

- CARLOS DELGADO KLOOS: Full Professor at Universidad Carlos III de Madrid, Director of the UNESCO Chair on "Scalable Digital Education for All", and Vice-President for Strategy and Digital Education. He introduced the teaching of Java at Universidad Carlos III de Madrid in 1997. Content creator and general supervisor of the MOOC.
- IRIA ESTÉVEZ-AYRES: Assistant Professor at Universidad Carlos III de Madrid. Content creator, instructional designer, and assessment designer.
- CARLOS ALARIO-HOYOS: Visiting Professor at Universidad Carlos III de Madrid. Content creator, instructional designer and responsible for the communication with learners.
- JORGE RUIZ: Part-time lecturer at Universidad Carlos III de Madrid. Content creator and quality controller
- JULIO VILLENA ROMÁN: Part-time lecturer at Universidad Carlos III de Madrid. Content creator and quality controller.
- RAQUEL M. CRESPO GARCÍA: Assistant Professor at Universidad Carlos III de Madrid. Content creator and responsible for laboratory activities.
- CARMEN FERNÁNDEZ PANADERO: Assistant Professor at Universidad Carlos III de Madrid. Content creator.

COURSE STRUCTURE

WEEK 1: *Linear data structures*

The first week starts refreshing arrays, which were explained for the first time in Part 1. Then, the concept of generics in Java is briefly introduced, this concept being of importance when working with data structures. Then, linked lists are presented, illustrating some of the basic operations that can be done with linked list. Finally, the Java API is explored to introduce some interfaces and classes that help us to work with linear data structures.

WEEK 2: *Stacks*

The second week addresses a case of linear data structure, stacks, where the insertion and extraction takes place in the same end. Array-based and linked-based implementations are introduced, studying the relationship between stacks and recursion before concluding the week.

WEEK 3: *Queues*

The third week addresses a case of linear data structure, queues, where the insertion and extraction takes place in different ends. Array-based and linked-based implementations are introduced. Special cases of double ended queues where insertion and extraction takes place in both ends, and priority queues, where elements are inserted or extracted in the queue depending on their priority are also studied.

WEEK 4: *Trees*

The fourth week focuses on non-linear data structures, and particularly trees, binary trees, binary search trees and heaps, presenting algorithms to work with them.

WEEK 5: *Sorting*

The last week studies six sorting algorithms (Bubble, Selection, Insertion, Heap, Merge and Quick Sort), comparing the time and number of operations they take.

COURSE METHODOLOGY

Every week follows the same methodology and structure. First, theoretical concepts are presented in videos and other audiovisual formats in a simple and pleasant way through examples and metaphors. The learner assimilates these concepts by practicing with formative (non-graded) exercises, receiving immediate feedback.

Next, a case study is introduced to demonstrate the theoretical concepts. The learner can download, analyze and modify the code of the case study to improve the understanding on the concepts taught. At the end of the week learners take the assessment activities (exams) that will be used to calculate the final grade.

Supplementary materials to delve into the topics of the course may be provided at any times.

The estimated time learners need to complete each week is from 5 to 7 hours.

COMMUNICATION AND SOCIAL COMPONENT

EMAILING and the **COURSE INFO PAGE** will be used by teachers to keep learners up-to-date with all the news related to the course.

In addition, **SOCIAL TOOLS** will be supported for learners to communicate with teachers and peers: the course forum on edX and Twitter (*#javaedxuc3m*). Programming can be very challenging, and difficulties will inevitably arise. Learners are encouraged to actively interact with other learners and teachers through these three social tools and share their concerns, problems, experiences and pieces of code.

EVALUATION

Evaluation will cover theoretical concepts and also small programs. These activities are mandatory only for those who wish to get a certificate at the end of the course.

The final grade for the course will be the result of the **FIVE EXAMS** (graded tests available only for verified track), each of them have a weight of 20%.

To **PASS THE COURSE** it will be necessary to obtain the 60% of the final grade.

CALENDAR

The course ***Introduction to Programming with Java 3: Fundamental Data Structures and Algorithms*** is now running and will be available until 30 June 2020 (23:59 UTC). Only learners in the verified track will be able to take the weekly graded assessments available until 30 June (23:59 UTC).

Certificates will be available on demand for learners as soon as they complete enough of the course with a high enough grade to qualify for a certificate. More information here.