

# Advanced Programming Methods

## Lecture 14 – RECAP

# Grading

Laboratory:--50%

- Lab Assignments (8 labs, each lab assignments is 5.5%, excepting Lab1 and Lab3 which are 1% each) -- 35%
- First Practical Test Exam at LAB5 : -- 5%
- Second Practical Test Exam: --10%

Final exam: -- 50%

- Written Exam (about 1 hour, closed books): --20%
- Practical Exam (about 4 hours, one problem in Java, one problem in C#, open books): --30%

Seminar:

- your contribution will be taken into account for the rounding of the final mark

# Rules

- in order to get into the final exam your average grade for the lab activities (assignments + 2 lab practical tests) must be at least 5
- in order to pass the final exam you must have at least 5 at the final written exam and at least 5 at the final practical exam such that the final grade must be at least 5
- you can pass either both the final written exam and the final practical exam or nothing
- those who attended the talk from 22DEC will get additional 0.5 points to the final grade (excepting the situation 4.5 to 5)

# Rules for the second exam ("restanta")

- in order to pass the final second exam you must have at least 5 at the final written exam and at least 5 at the final practical exam
- you can pass either both the second final written exam and the second final practical exam or nothing
- at the second exam "restanta" you can present or re-present maximum 5 lab assignments in order to improve your lab activities mark
- in order to be able to present or re-present the lab assignments you must have to pass the second exam
- the second exam grade will automatically replace the first exam grade

# Lecture 1-Java

- Java syntax
- OO concepts
- classes, objects, constructors, this, methods, passing parameters
- overloading
- array
- static methods and fields

# Lecture 2 -- Java

- code reusing
- inheritance, method overloading, protected
- superclass, super
- field initialization
- method overriding
- class Object
- polymorphism
- abstract classes
- interfaces
- template method pattern

# Lecture 3 -- C#

- basics, arrays
- classes, constructor, destructor, finalize, methods, properties, indexers, inheritance, virtual methods, base, sealed, static constructors, static methods, static classes, partial classes, as, is, class object
- struct
- operator overloading
- namespaces

# Lecture 4 -- C#

- Delegates
- Implicitly Typed Local Variables
- Lambda Expressions
- Extension methods
- Events (Observer pattern)



# Lecture 5- Java+C#

- Exceptions in Java and C#
- Java packages
- C# namespaces

# Lecture 6 -- Java

- Generics: autoboxing, generic methods, generic arrays, bounds, wildcards, bounded wildcards
- Collections:

# Lecture 7 -- C#

- generics, c# vs java, generics types, generics methods, generics constraints

- collections

# Lecture 8 java and c#

- File operations in java
- file operations in c#

# Lecture 9 --Java

## Java GUI

# Lecture 10 -- C#

- C# GUI

# Lecture 11-Java

## Concurrency – basics, threads vs processes

- What is a thread ?
- Define and launch a thread
- The life-cycle of a thread
- interrupt a thread
- thread synchronization
- java monitor
- threads group

# Lecture 12 -- C#

- threads
- threads life cycle
- join, sleep
- creating and starting threads
- foreground and background threads



# Lecture 13 -- C#

- thread pool: via the Task Parallel Library, by calling `ThreadPool.QueueUserWorkItem`, via asynchronous delegates
- synchronization: basics, locking, deadlocks, mutex, semaphores, signaling