CM4106 - LANGUAGES & COMPILERS

MODULE INTRO

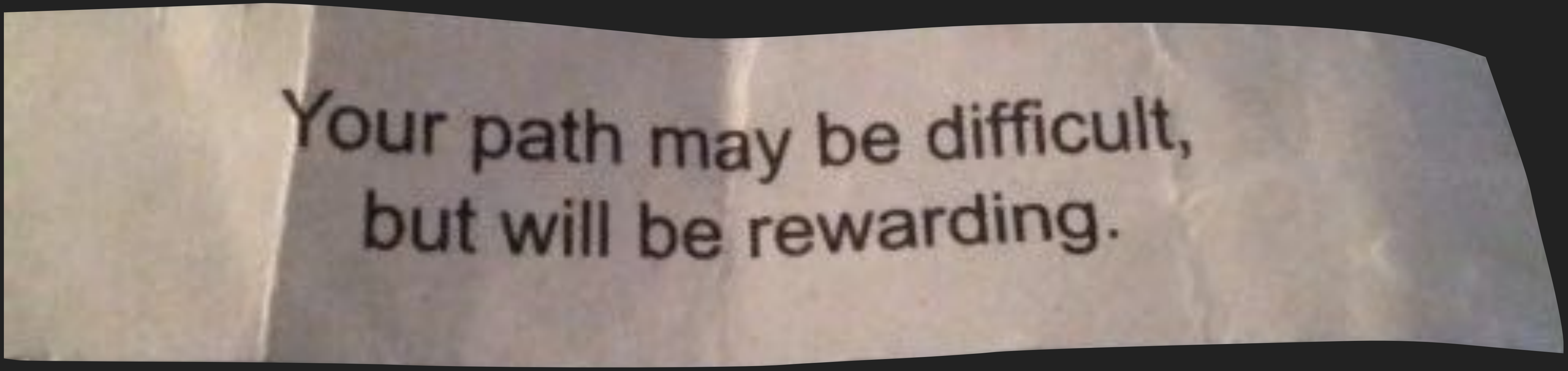
WELCOME TO HONOURS YEAR

- This is it, the big one.
- I doesn't matter how you got here
- this is the only year that counts towards your classification!

MODULE WELCOME

- Languages and Compilers
- Module Aim - to help you understand how;
    - Programming languages are defined
    - How compilers translate between languages
    - How compilers can be used to generate code

DIFFICULTY?



Your path may be difficult,
but will be rewarding.

Fortune Cookie - The Golden Palace - circa 2003

- Language definition is complex
- Compilers are technical
- But both will give you a better understanding of how your code is compiled.

CLASSES

- Lecture - Thursday 9-10 - N344
- Lab - Friday 9-12 - N528
- This is the first time this module has ran, so we may look at how this works out over the next few weeks

- Me - j.p.isaacs@rgu.ac.uk
- Main staff office - 01224 262789

ASSESSMENT

- Coursework
  - Compiler Development 50%
    - Build a compiler for a given language which will generate code for a given target machine
  - Exam 50%
    - Language and Compiler theory

WHY THIS MODULE

- Everything that your computer does is the result of a program.
- Each of these programs has been specified in a language and compiled to code that runs on the target machine
- Example target machines: JVM, .NET, Unreal Engine, Unity
    - Think about scripting character movement in a game, or the parameters for an algorithm

REAL WORLD EXAMPLE

```
START
MOVE LEFT 20 DEGREES,
LOWER 30 DEGREES,
PICKUP OBJECT
END
```

Non Expert “code"

compile to some intermediate

```
initialiseArm(){ … run(commandlist);
shutdown() …}

move(degrees){…}

lower(degrees) {…}
```

Executed to target

WHAT ARE COMPILERS WRITTEN IN THEN?

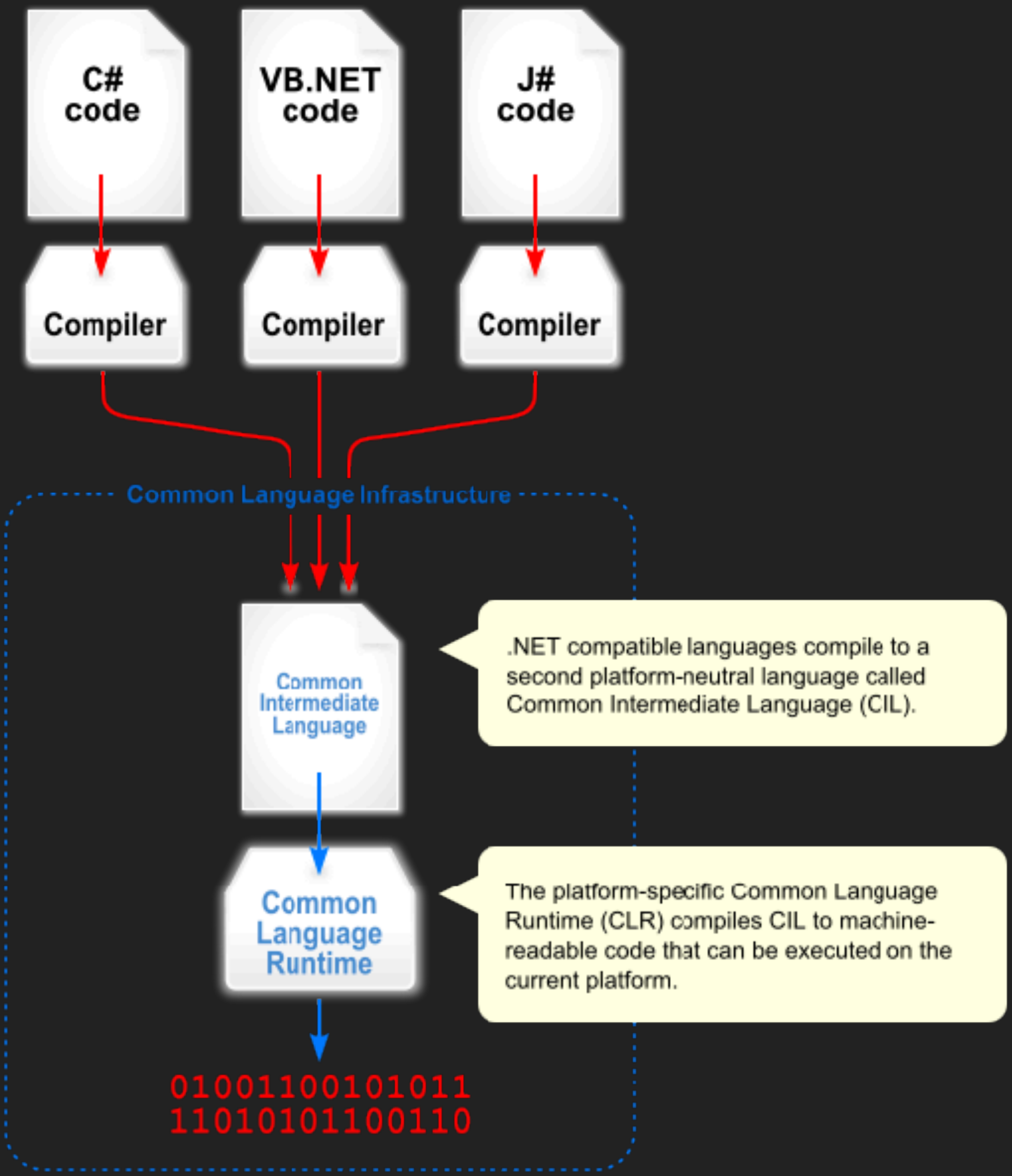
- A compiler takes one language and translates it into the language or executable of the target machine.
- But what is the compiler it’s self written in?
  - Answer - Anything!
    - The JAVAC java compiler is written in C
    - The RUBY compiler is mainly written in java
    - Roslin (a new c# compiler) written in C# (kind of)

WHAT ARE WE WRITING OURS IN?

- Most of you, particularly in at RGU will not have had experience of the .NET languages, specifically C#.
- We wanted to give you that experience.
- So we will be writing our compilers in C#
- More on our Source and Target next week.

.NET LANGUAGES

- Suite of languages that use specific compilers
- Each of the compliers compiles the code to the CIL
- Then the CLR which can be executed on a range of target platforms.



.NET compatible languages compile to a second platform-neutral language called Common Intermediate Language (CIL).

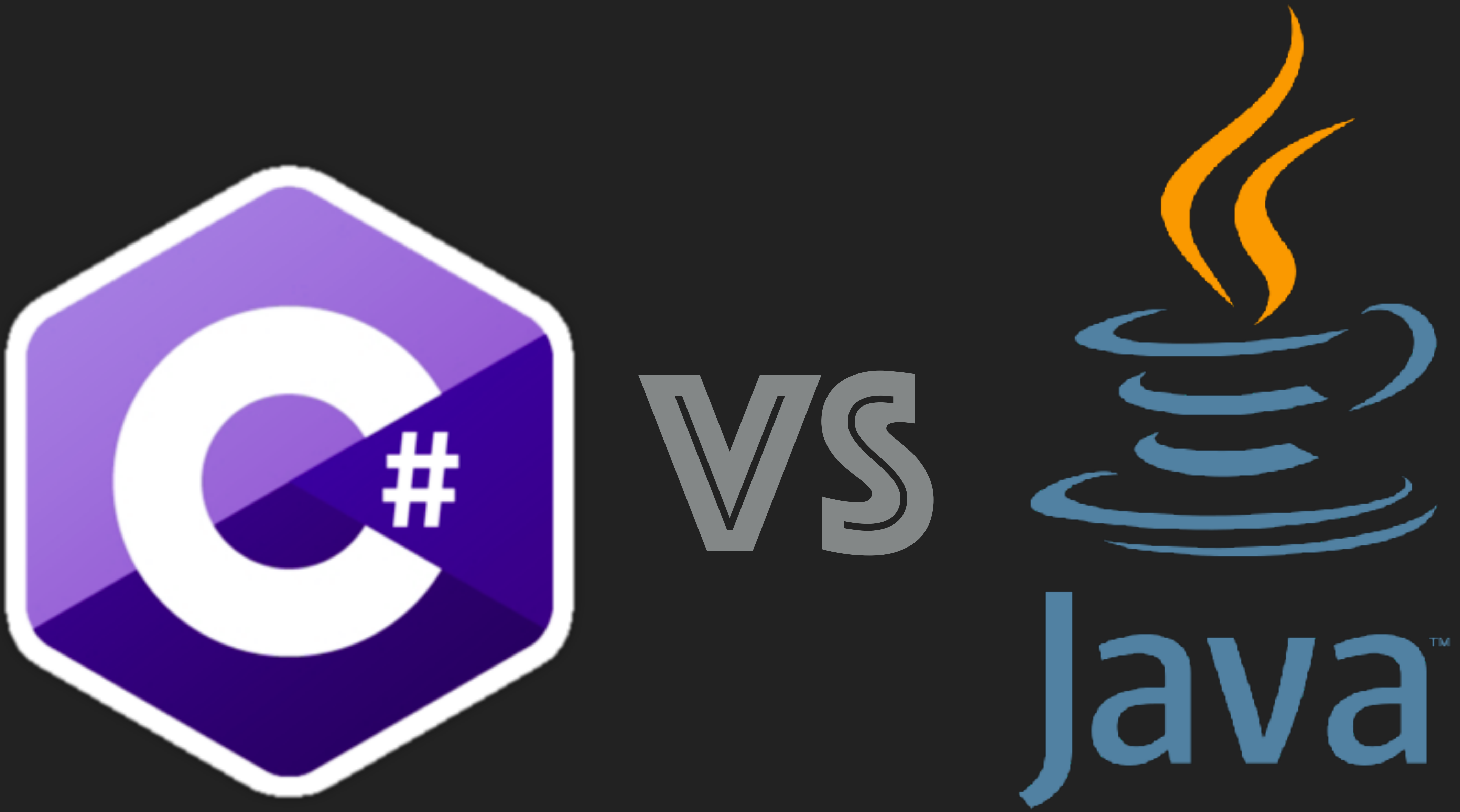
The platform-specific Common Language Runtime (CLR) compiles CIL to machine-readable code that can be executed on the current platform.

C#

- C# is a general purpose object orientated programming language
- Was created from the ground up in an attempt to over come some the perceived issues with JAVA
- It was believed by the developers that JAVA had become bloated and over designed as it developed.

- C# used to have a mascot called Andy but they retired it in 2004, can't think why.

C# VS JAVA



WHAT'S THE SAME

- Both are fully object orientated
  - Everything is an Object
  - Objects are references
  - dot notation used for referencing fields and methods
    - e.g. player.score or player.fire(bullet)

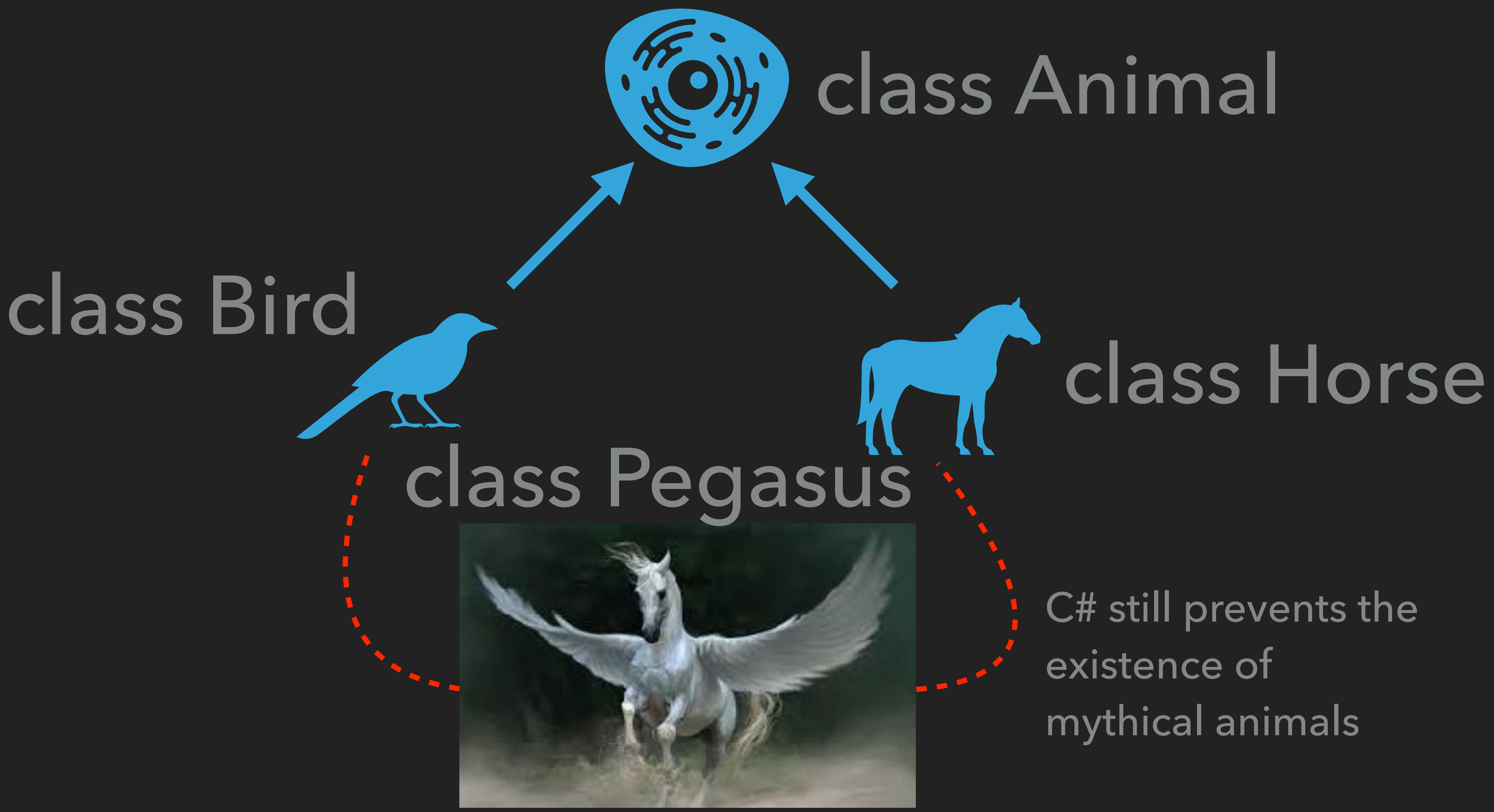
GARBAGE COLLECTION

- Objects are created on the HEAP.
- For an object to exist it has to have a reference
- If it doesn't it's removed.
- `BadaBing badaBoom = new BadaBing();`
- `badaBoom = null;`

MORE OF THE SAME

- Both C# and JAVA are type safe / type strict
  - e.g an int is always an int unless you convert or cast it.
  - both will throw exceptions at run time if this is broken.

SINGLE INHERITANCE



class Animal

class Bird

class Horse

class Pegasus

C# still prevents the existence of mythical animals

WHAT'S DIFFERENT

- the main differences that you need to worry about are syntax.
- I'll go through the main ones in the next few slides.
- C# is a complex language

"It seems as if Java was built to keep a developer from shooting himself in the foot. It seems as if C# was built to give the developer a gun but leave the safety turned on. "

IMPORTING LIBRARIES

- JAVA - `import` keyword
  - import java.util.*;
  - import java.io.*;
- C# - `using` keyword
  - using System;
  - using System.IO;
  - using System.Reflection;

APPLICATION ENTRY POINT

- Like Java, C# applications can only have a single entry point i.e the main method.
- JAVA -
  - public static void main (String[] args)
- C# -
  - static void **M**ain(**s**tring[] args)
  - static void **M**ain()

PRINTING TO CONSOLE

- JAVA -
  - System.out.println(“Here’s Jonney”)
- C# -
  - System.Console.WriteLine(“Here’s Jonney”)
  - Console.WriteLine(“Here’s Jonney”)

CONSTANTS

- JAVA -
  - Static final int meaning = 42;

Compile time constants, value set at compilation time

cannot be changed

- C# -
  - const int meaning = 42;
  - readonly int meaning = 42;

Runtime constant, value set once during program

PRIMITIVE TYPES

- JAVA -
  - Signed byte, short ,int and long
- C# -

byte	ubyte	8
ushort	short	16
uint	int	32
ulong	long	64
decimal	holds decimals without rounding	

SWITCH STATEMENTS

```
switch(foo){
  case "A": Console.WriteLine("A seen”);break;
  case “B":
  case "C": Console.WriteLine("B or C seen");break;
   /* ERROR: Won't compile due to fall-through at
  case "D" */
  case "D": Console.WriteLine("D seen”);
  case "E": Console.WriteLine("E seen");break;
}
```

- C# -
  - Breaks are mandatory, unless the case is empty (no fall through)
  - Unlike JAVA you can switch any type, not just primitives.

ENUMERATIONS (ENUMS)

- JAVA -
  - Code:
    ```
    public class Direction {
              public final static int NORTH = 1;
              public final static int EAST = 2;
              public final static int WEST = 3;
              public final static int SOUTH = 4;
              }
    ```
  - Usage: `int wall = Direction.North;`

This is not type safe, as any int could be given to the wall

- C# -
  - Code: `public enum Direction {North=1, East=2, West=4, South=8};`
  - Usage: `Direction wall = Direction.North;`

This is type safe because only specified values can be given

FOREACH

- JAVA does not have this (although javascript does)
- A simple way if iterating through anything which can be iterated (implements Enumerable).

```
string[] greek_alphabet = {"alpha", "beta",
  "gamma", "delta"};

 foreach(string str in greek_alphabet) {
  Console.WriteLine(str + " is a greek letter");
}
```

PROPERTIES

- C# provides convenient encapsulation (getters and setters)

- JAVA

```
private int size;

public int getSize(){

   return size;

}

public void setSize (int val){

   size = val;

}
```

- usage
- int x = someclass. getSize();

- C#

```
private int size;

public int Size
{
   get {return size;}
   set {size = val;}
}
```

- usage -
- int x = someclass.Size();

INHERITANCE

- JAVA -
  - class Pegasus **extends** Bird **implements** canFly

Java defines inherited classes and interfaces

- C# -
  - class Pegasus **:** Bird, canFly

Everything defined by single : then the classes and interfaces being extended

POLYMORPHISM & OVERRIDING

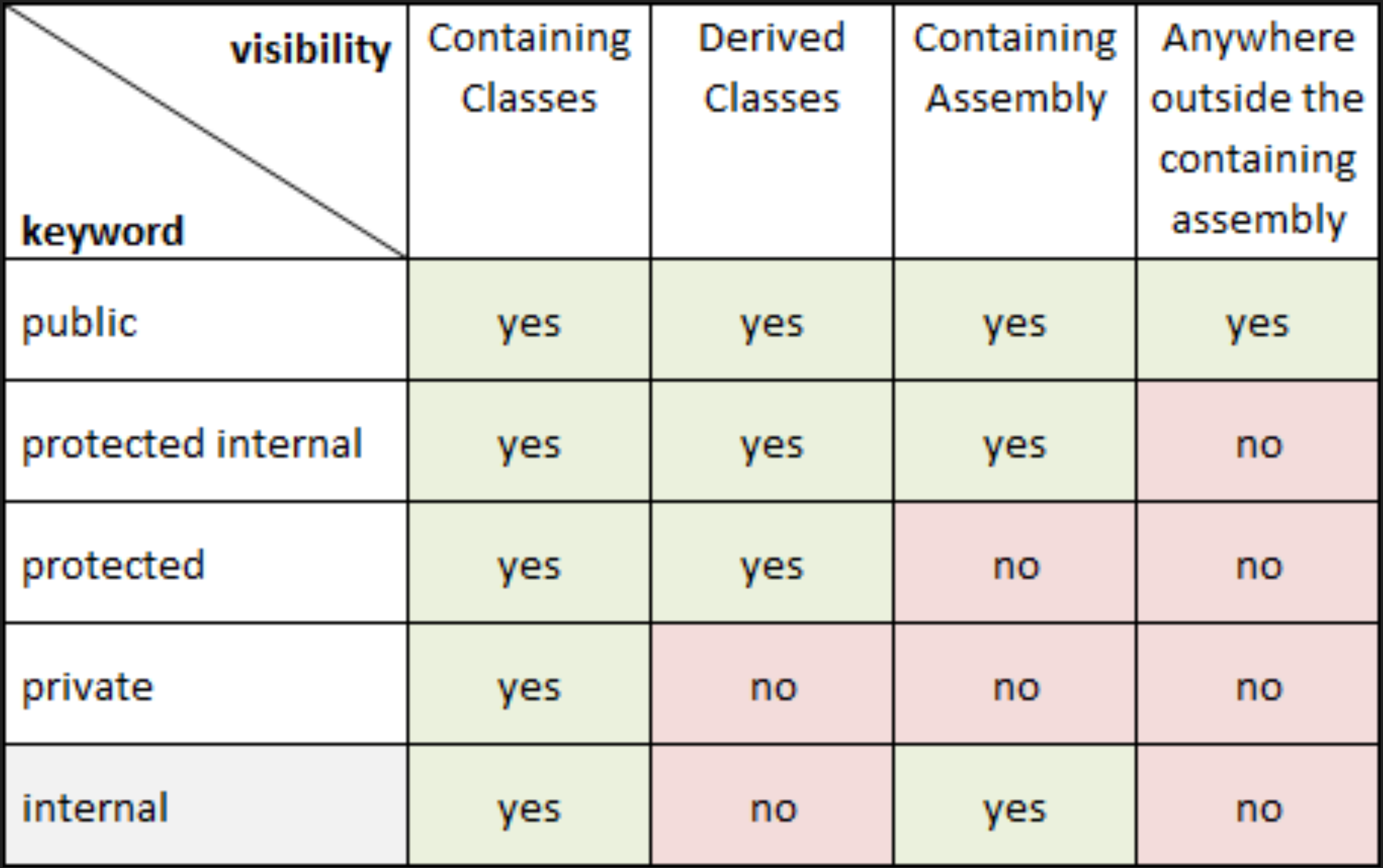
- C# - Methods that will be overridden must be marked as virtual

```
using System;

public class Parent{
   public virtual void DoStuff(string str){
      Console.WriteLine("In Parent.DoStuff: " + str);
   }
}

public class Child: Parent {
   public void DoStuff(int n){
      Console.WriteLine("In Child.DoStuff: " + n);
   }
   public override void DoStuff(string str{
      Console.WriteLine("In Child.DoStuff: " + str);
   }
}
```

ACCESS MODIFIERS



visibility keyword	Containing Classes	Derived Classes	Containing Assembly	Anywhere outside the containing assembly
public	yes	yes	yes	yes
protected internal	yes	yes	yes	no
protected	yes	yes	no	no
private	yes	no	no	no
internal	yes	no	yes	no

NAMESPACE VS PACKAGE

- JAVA - Package vs C# - Namespace
  - Create scope to avoid collisions and to group similar classes essentially the same purpose
  - C# simply namespace MySpace { classes_go_here }

67%
2
Tests passed
Sign in
Sign in
Visual Studio

VISUAL STUDIO

- Being a .NET language C# is usually developed using the Visual Studio IDE
- Because you are students you can get it free from MS imagine (previously Dreamspark)
- There are editions for both Mac and PC, it’s also in the labs.
- Todays lab will mainly be about getting used to it

SUMMARY

- C# is a new language for most of you
- It is similar to JAVA - with some syntax differences
  - there is a cheatsheet on Moodle that will help you get started
  - We will start of slowly before speeding up to compiler development.