

# **INTRO TO COMPUTER PROGRAMMING II**

A.K.A. CS2040, SPRING 2020

# Course Content

- ▶ Object Oriented Programming (OOP)
- ▶ Programming language: Java
  - ▶ Classes and Objects;
  - ▶ Inheritance, Exceptions, Polymorphism;
  - ▶ JavaFX (Graphics);
  - ▶ Event Driven programming, I/O, Events;
- ▶ Useful notions: Recursion and Complexity

Course web page: <http://matteoacclavio.com/Course.html?CS2040-2020>

# How to learn programming?

- ▶ Attend classes (fruitfully);
- ▶ Study the book (Y. Daniel Liang, Introduction to Java programming);
- ▶ PRACTICE, PRACTICE, PRACTICE:
  - ▶ MyProgrammingLab;
  - ▶ Home assignments;
  - ▶ Understand the errors;
- ▶ Whenever you don't know how to do, try to ask (in the order):
  - ▶ THE internet (search if somebody had the same problem);
  - ▶ Tutor Gavin Goerke ( [a101053@aup.edu](mailto:a101053@aup.edu) )
  - ▶ Teacher (Office hours: after classes, by appointment)

# Course evaluation

Accordingly with the “how to learn”:

- ▶ 20% - Online exercises
- ▶ 40% - Exercises
- ▶ 15% - Midterm Exam (test and discussion about corrections)
- ▶ 25% - Final Project (develop a really tiny program with graphical interface)

Attendance is mandatory

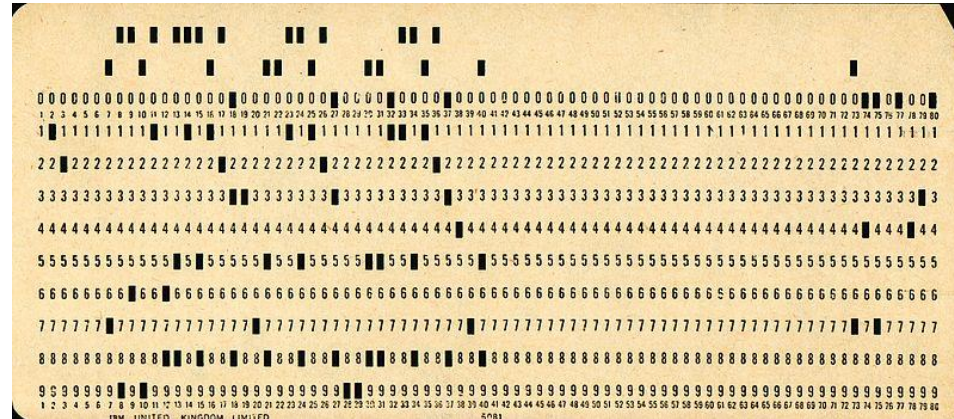
Assignments are mandatory for learning... but also to pass the exam

# MyProgrammingLab

- ▶ Exercises in line with the course progression;
- ▶ Instant feedback;
- ▶ Course ID: PEAR-41701-UUNW-43
- ▶ <http://myprogramminglab.com>

# How to program?

- ▶ Machine / Assembly Language
- ▶ Programming language
- ▶ Compilers
- ▶ IDE



# About programming

- ▶ Errors during programming are the routine;
- ▶ It is important to understand errors and correct them;
- ▶ Type of errors:
  - ▶ Compilation errors;
  - ▶ Execution errors;

# PROGRAMMER GOLDEN RULES

**DOT = Do One Thing**

Each function carry out one task.

More tasks at the same time increase the probability to mess with the code!

**DRY = Don't Repeat Yourself**

If the same task occurs multiple times, then create a function to do it.

Write many times the same function dealing the same task is not optimal and increases the probability to mess with the code!

**KISS = Keep It Stupidly Simple**

Each line of your code do only one simple task.

Guess what may happen by writing a lot of functions in the same program line...



# Data Type and Program Type

- ▶ Each data has a type;
- ▶ Some elementary Data Types in Java:
  - ▶ byte (Integer number);
  - ▶ int (Integer number);
  - ▶ boolean (Boolean/conditional);
  - ▶ float (Rational number);
  - ▶ char (Unicode character);
- ▶ Stings and Arrays are not elementary data type;

# Some Unicode character codes

	0	1	2	3	4	5	6	7
0	NUL	DLE	space	0	@	P	`	p
1	SOH	DC1 XON	!	1	A	Q	a	q
2	STX	DC2	"	2	B	R	b	r
3	ETX	DC3 XOFF	#	3	C	S	c	s
4	EOT	DC4	\$	4	D	T	d	t
5	ENQ	NAK	%	5	E	U	e	u
6	ACK	SYN	&	6	F	V	f	v
7	BEL	ETB	'	7	G	W	g	w
8	BS	CAN	(	8	H	X	h	x
9	HT	EM	)	9	I	Y	i	y
A	LF	SUB	*	:	J	Z	j	z
B	VT	ESC	+	;	K	[	k	{
C	FF	FS	,	<	L	\	l	
D	CR	GS	-	=	M	]	m	}
E	SO	RS	.	>	N	^	n	~
F	SI	US	/	?	O	_	o	del

# Type Casting

- ▶ Type casting allows to read a data of a given type as a data of a different one.

For example: each character is saved in the computer as a number.

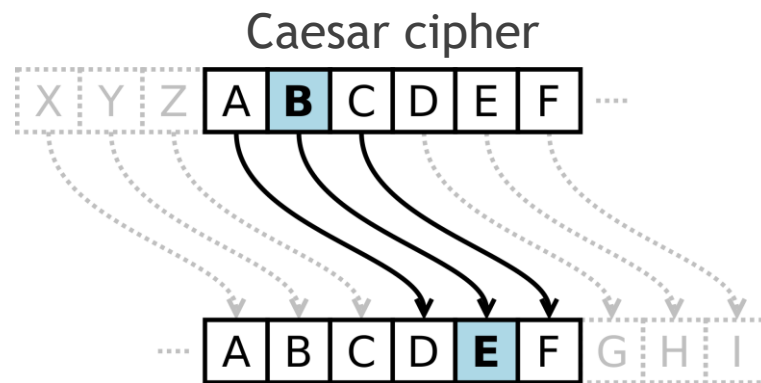
Excercise:

write a function which prints the character with Unicode a given integer i

```
static void printCharWithUnicode(int i)
```

# Cryptography

- ▶ Cryptography is required for secure communication
- ▶ The idea: you need a key to read the message
- ▶ “Famous” cryptography stuff: Caesar cipher, Enigma (WWII), HTTPS, RSA,...



# A simple encryption algorithm

We will write a program to encrypt messages by using a similar (but safer) encryption method based on a **stream cipher**

What we need:

1. Decompose a text into an array of characters
2. Transform each character into a number
3. Encrypt each number with our key

# Warm up exercise: Caesar cypher

- ▶ Write a function which transform a string to an array of characters

For example:

```
myString = "Caesar"
```

```
myArray=[C, a, e, s, a, r]
```

- ▶ Create an array of integers containing the Unicode of each character in myArray
- ▶ Fix a shift (an integer n) and write an encryption function
- ▶ Convert back each Unicode into a character
- ▶ Print the encrypted message