MINISTÈRE DE LA DÉFENSE

# Engineer semester

## “Basics of digital information processing and protection”

A mathematical and computer science point of view.

ÉCOLES DE SAINT-CYR COËTQUIDAN

## COURSE DESCRIPTION

### I GENERAL INTRODUCTION

In contemporary science some areas at the border between mathematics and computer science are growing fast and appear to become crucial to engineers dealing with applications. Some of the future army officers will have to communicate with high level scientists. We must enable these officers to maintain such exchanges in order to keep military forces up to date in mastering modern weapons.

Our educational program is geared towards the intersection of mathematics and computer science. We chose promising topics with military application in mind.

We will give an introduction to mathematical logic and functional programming. These two areas are known to be closely related since the proof-program correspondence was established during the 1970-1980s. They have had a strong impact on industry ever since. On the one hand critical software requires safety and security assurance at design time. On the other hand all the modern programming languages have moved to multi-paradigm approaches to benefit from the intrinsic recognized qualities of functional programming. Such programming languages are currently rarely used in the military domain. Still the current trend towards cyber defense will increase the need for proven security in the design of software systems. Logic and functional programming are invaluable tools to satisfy such needs like it has been the case for the safety of critical systems. Moreover mathematical logic (which is part of traditional philosophical education in many countries) is supportive for decision making: it helps to design rational reactions to specific situations.

We will teach an introductory probability theory course mainly oriented towards discrete probability but including random numbers (a key aspect of cryptography) and introduction to Markov chains (useful to master the basics of information theory).

A large time will be devoted to modern cryptography: its mathematical foundations (algebra, number theory, information theory) as well as its different kinds of algorithms. Students will have to learn programming. Taking into account the main drift of the curriculum, C or C++ would be a highly suitable choice as programming language, as well as the use of mathematical software like SAGE. So our students will be able to experiment with different algorithms introduced during several courses. The emphasis given to cryptographic techniques in our semester has obvious reasons: protecting and obtaining information are vital matters in modern (cyber) war.

An introduction to quantum computing is also included in the program. This new and probably promising area deserves some consideration, given its potentially enormous impact on cryptography. Last but not least we decided to include in our program a slight contribution from humanities: that seems to us to be a "must" to complete a scientific education.

This program was designed by numerous exchanges involving people from different universities and military academies and institutions: Direction Générale de l'Armement, Universität der Bundeswehr München, Centro Universitario de la Defensa (Zaragoza), Université de Rennes (Institut de Recherche Mathématique de Rennes), IRISA (Institut de  Recherche en Informatique et Systèmes Aléatoires), Université de Rouen (computer science) and of course Écoles de Saint-Cyr Coëtquidan (mathematics and computer science).

II Hours and European Credits Transfer System (ECTS)

| | | **Hours** | **ECTS** |
|---|---|---|---|
| Randomness, information, computation | | 80 | 6 |
| | Probability theory | 40 | |
| | Information theory | 20 | |
| | An introduction to quantum computing | 20 | |
| | | | |
| Logic, proofs, programming | | 90 | 6 |
| | Functional programming | 50 | |
| | Introduction to mathematical logic and lambda calculus | 40 | |
| | | | |
| Cryptology: Theory and Practice | | 100 | 10 |
| | A short introduction to algebra | 20 | |
| | Proofs of security | 20 | |
| | Computational number theory and public key cryptography | 20 | |
| | Design, security of cryptographic algorithms, cryptanalysis | 20 | |
| | Algorithms and Programming in C++ and SAGE | 20 | |
| | | | |
| Humanities | | 44 | 3 |
| | Leadership for officers | 22 | |
| | Geopolitics of war | 22 | |
| | | | |
| French language | | 60 | 3 |
| | | | |
| French defence policy | | 18 | 2 |
| Tactical and historical studies | | 22 | |
| | | | |
| Personal studies | | 42 | |
| | | | |
| Semester presentation | | 2 | |
| | | | |
| **Total** | | **458** | **30** |

III MODULE DESCRIPTIONS

# Logic, proofs, programming

## INTRODUCTION TO MATHEMATICAL LOGIC AND LAMBDA CALCULUS (40 hours)

### 1. Introduction

Logic is a typical tool that helps making decisions thanks to modeling and reasoning about situations. Logical languages allow modeling propositions as formulae and terms. Then logic gives a meaning to the formulae. Namely, natural deduction and sequent calculus provide formal and well-founded semantics to the intuitive notion of deduction. Altogether, these tools provide a framework to build and check proofs with the assistance of computer software.

Since the 70s and 80s, logic and functional programming are closely entangled. On the one side the Curry-Howard correspondence states that proofs and programs are similar objects. Indeed inference rules of natural deduction or sequent calculus are similar (if not identical) to the type checking rules of functional languages. So proofs turn out to be correct-by-construction executable programs. This is an approach to build reliable, safe and secure software. On the other side, the lambda calculus defines a family of semantics for functional programming. Thanks to data immutability, this calculus is far simpler than other calculus with side effects. Functional programming thus offers a simpler setup to the introduction to program verification.

### 2. Table of contents

- First order logic
- Language, terms, formulas
- Proofs
- Natural deduction
- Sequent calculus
- The terms of lambda calculus
- Substitution and beta conversion
- Types
- Typed lambda calculus
- Relations to the Coq language

### 3. References/sources

Adam Chlipala
Certified Programming with Dependent Types
Publisher: MIT Press
Dirk van Dalen
Logic and Structure
Publisher: Springer

J. Roger Hindley, Jonathan P. Seldin
Lambda-Calculus and Combinators: An Introduction
Publisher: Cambridge University Press

J. L. Krivine
Lambda-Calculus, Types and Models
Publisher: Ellis Horwood Ltd

**FUNCTIONAL PROGRAMMING (50 hours)**

**1. Introduction**

Functional programming is paradigm and programming style that relies mainly on the concept of function. In the industry, it is mainly used in specific niches that include critical systems, one of the most famous ones being the trading system of Jane Street Capital. Since more than a decade, most of the modern programming languages like Java or C# tend to evolve, becoming multiparadigm languages that borrow several principles and features from functional programming. The popular Map Reduce parallel computing framework is also a typical system inspired by common functional programming idioms.

Functional programming relies on few simple yet powerful concepts. Functions are at the center of the approach. As first-class values, functions can be used as parameters and return values as well, which are at the core of frameworks like Map Reduce. Data immutability avoids side effects, which improves reliability. The programming style is mainly based on recursive data structures and algorithms along with pattern matching, which typically provides equational and declarative notations.

In addition to its intrinsic interest, functional programming is also a key prerequisite for the introduction to logic and lambda calculus. Indeed, the latter defines the family of semantics for functional programming, far simpler than other programming paradigms thanks to data immutability. The Curry-Howard correspondence makes functional programming an approach of choice in proof assistants.

**2. Table of contents**

- Recursive algorithms
- Recursive data types
- Pattern matching
- Polymorphic types and polymorphic functions
- Higher order functions
- Modules and functors

**3. References/sources**

Guy Cousineau, and Michel Mauny
The Functional Approach to Programming
Publisher:  Cambridge University Press

Adam Chlipala
Certified Programming with Dependent Types
Publisher: MIT Press

## Randomness, information, computation

**PROBABILITY THEORY (40 hours)**

**1. Introduction**

No event can be predicted with total certainty. The best we can say is how likely they are to happen, using the idea of probability. However almost nobody can deal with randomness. The intuition of almost everybody is wrong when it comes with probabilities. Even very smart mathematicians had made big mistakes on very simple random problems. The reason is that, since we are born, we are taught to think in a deterministic way. Although every aspect of our lives is random, we treat it as deterministic.

That is why we will study the probability theory in the discrete case in order to bring the cadets to reason in a probabilistic way. They should question their way of thinking. To do this, we will treat many simple cases and paradoxes. The question of the true nature of randomness will be dealt with.

Indeed, although randomness appears everywhere and all the time, it seems that randomness doesn't exist (except maybe at quantum level).

Then we will focus on Markov chains. In 1907, A. Markov began the study of an important new type of chance process. During an experiment whose such a process is a model, the next outcome is affected by the current one but is not by the past ones. This type of process is called a Markov chain. It has application in every aspects of science. We will study some powerful results of such processes.

Finally, we will deal with Pseudo Random Number Generators (PRNG). Such generators create numbers that are "random" like. PNRG are central in applications such as simulations (e.g. for the Monte Carlo method) and cryptography. The question of how "random" like those PRNG are will be discussed. It will imply a statistical study. This will lead us to develop basic statistical concepts.

This 40 hours course will be organized around three distinct parts: the probability theory in the discrete case, Markov chains and the study of pseudo-random generators.

## 2. Prerequisites

Cadets should get used to manipulate basic mathematical notations;
use abstractions to describe concepts;

have familiarity with calculus:
- variables and functions (including integration and derivation),
- sequences and limits,
- convergent and divergent series;

manipulate basic elements of counting:
- combinatorial elements,
- ordered samples,
- factorials,
- binomial coefficients,

have familiarity with basic set theory:
- sets and subsets,
- set relations,
- unions, intersections, complements, differences,
- de Morgan’s laws.

## 3. Table of contents

1. Discrete Probability
Set Notation and Probability,
Random Variables and basics law,
Expected Values,
Random Vectors,
Conditional Probability,
Asymptotic law.

2. Discrete Markov chains
Definition and basic properties,
Transition Matrix,
Classification of States,
Irreducible Chains.

3. Pseudo Random Number Generator and Statistics
Pseudo randomness,
Examples of PRNG,
Statistics and PRNG.

**4. References/sources**

William Feller
An Introduction to Probability Theory and Its Applications, Vol. 1,
Publisher: Wiley

Donald E. Knuth
Art of Computer Programming, Volume 2: Semi numerical Algorithms
Publisher: Addison-Wesley Professional

**INFORMATION THEORY (20 hours)**

**1. Introduction**
Since the famous Shannon paper from 1948, information theory has become the scientific field supporting the engineering of telecommunications. For instance, the capacity is the good measure of the quality of a transmission over a wireless channel, and the decoding capability is the good measure of the quality of a telecommunication system adapted to the given transmission channel.
In cryptology also, since the Shannon paper from 1949, information theory is a fundamental domain.
- for key generation, Shannon entropy is the good measure of the average randomness of a source
- in the design of symmetric primitives, Shannon principle of diffusion (spreading randomness) is implemented through shift registers and more recently MDS matrices.
- in the analysis of symmetric primitives through statistical cryptanalysis, correlation attacks.
- in the analysis of a telecommunication chain in interception context

During this course some essential notions of information theory will be introduced which will give the student the basics to be able to follow the course entitled DESIGN, SECURITY OF CRYPTOGRAPHIC ALGORITHMS, CRYPTANALYSIS. We will terminate the course by an example of how to use the information theory principles to reconstruct a digital telecommunication chain in a non-cooperative context.

**2. Prerequisites**

Basic notions of probabilities (mean, variance, independence of variables,...)
Notions of linear algebra (matrices, vector spaces)
Notions of Boolean algebra, polynomial rings, Euclidian rings
Algorithmic and complexity notions

**3. Table of contents**

- Compute the capacity of some channels
- Compute the entropy of some sources
- Be aware of the problematic of error-correction coding
- Understand the problem of decoding complexity
- Entropy
- Channel capacity
  - Definition
  - Binary symmetric channel (BSC)
  - Binary erasure channel BEC
  - Binary input additive white Gaussian noise channel (BIAWGN)
  - Shannon theorem

- Decoding algorithms
- Complexity of decoding
- Information set decoding
- Iterative decoding
- Ad hoc decoding
- Information recovery (code reconstruction, scrambler reconstruction)
- Framework
- Algorithms

### 4. References/sources

Todd K. Moon
"Error Correction Coding"
Publisher: Wiley-Interscience

A. Barg "Complexity issues in coding theory", Handbook of Coding theory chap 7

## AN INTRODUCTION TO QUANTUM COMPUTING (20 hours)

### 1. Introduction

The technological advances since the invention of the microprocessor (1970) to the most recent achievements (2015) have been exponential, leading to massive integration of transistors on a given microchip. In the last generation of microchips the individual transistors are separated by just 100 interatomic distances of the Silicium substrate. Such a pace of technological advance cannot be sustained since we shall reach soon the ineluctable barrier of atomic scale lying in the quantum realm.
Quantum computing, and more generally quantum information, instead of considering the infrangible quantum barrier as an obstacle, takes advantage of the tremendous possibilities offered by the quantum world to encode, process, transmit, and protect information by using quantum protocols. Quantum communication and cryptography are already developed in a pre-industrial level.
Quantum computing is still at a putative level; quantum processing devices have been constructed as proofs of principles. The course, after the foundational setting of quantum mechanics and there call of basic mathematical tools, will introduce q-bits, the quantum generalization of classical bits. Then the notion of quantum gates and quantum circuits will be developed. A series of quantum algorithms will then be described, culminating with the celebrated Shor’s factoring algorithm. For a physical process to be useful as a computing algorithm, its ability to correct errors is a pre-requisite. Hence the main quantum error correcting algorithms will be reviewed.
The course will end with a choice among the following topics: either some theoretical considerations concerning quantum Turing machines and quantum complexity classes or with some topics in quantum communication such as teleportation, dense coding, etc.

### 2. Prerequisites

Basic acquaintance with finite dimensional complex vector spaces and linear and bilinear algebra (including quadratic and hermitian forms) on them.
Note that no prior knowledge of quantum mechanics is required as the essential notions will be introduced during the course.

### 3. Table of contents

1. Brief history of computing technology: computing is a physical process.
The quantum boundary.
2. Basic postulates of quantum mechanics with an emphasis on the measurement postulate.
13. Short résumé of mathematical tools with an emphasis on the notion of tensor products of Hilbert spaces. Entanglement and partial measurements.

4. From c-bits to q-bits: reversible gates and quantum circuits.
5. Basic quantum algorithms: Deutch-Josza parallelization, quantum Fourier transform, Grover's quantum search, Short's quantum factoring algorithms.
6. Quantum error correction: quantum repetition codes, Shor's code, Calderbank-Shor-Steine code.
7. A choice among: quantum Turing machines and quantum complexity
classes or quantum communication.

**4. References/sources**

• Michael A. Nielsen and Isaac L. Chuang, Quantum computation and quantum information, Cambridge University Press, 2000.

• Emmanuel Desurvivre, Classical and quantum information theory, Cambridge University Press, 2009.

• N. David Mermin, Quantum computer science, Cambridge University Press, 2007.

• Yuri I. Manin, Classical computing, quantum computing, and Shor's factoring algorithm, Séminaire Bourbaki, 41:375–404, 1999.

• A. Yu. Kitaev, A. H. Shen, and M. N. Vyalyi, Classical and quantum computation, American Mathematical Society, 2002.

• Noson S. Yanofsky and Mirco A. Mannucci, Quantum computing for computer scientists, Cambridge University Press, 2008.

## Cryptology: Theory and Practice

**ALGORITHMS AND PROGRAMMING IN C++ AND SAGE (20 hours)**

**1. Introduction**

The purpose of the course is to explain how to program some algorithms using C++ or Python. For example (non restrictive list) algorithms for
Classification: A Markov distinguisher recognizes the language of a text
Simulation: Generating random numbers following a given law of probability Simulating an Linear Feedback Shift Register (LFSR)
Cryptography: Hashing and collisions The primality testings (from Fermat to Rabin Miller tests)
Factorizing big numbers (related to Rivest-Shamir-Adleman (RSA) algorithm)
Images: Examples of image processing

**2. Prerequisites**

Objects, Types, and Values
Texts, streams, numeric
Functions
Errors and exceptions

**3. Table of contents**

The Standard Library (STL) : Vectors, Templates, and Exceptions, Containers and Iterators
The Basics of Python: Python as a "wrapping" language

The basics of Sage Math toolbox: Introduction to the mathematical toolbox
Using Sage Math software for investigations on prime numbers, elliptic curves cryptography

**4. References/sources**
An excellent book on C++ is:
Bjarne Stroustrup, Programming -- Principles and Practice Using C++, second edition, Addison Wesley

C++: A classical free compiler and its documentation at https://gcc.gnu.org/
Python: Downloads at https://www.python.org
Sage Math: The official website at http://sagemath.org/index.html

## A SHORT INTRODUCTION TO ALGEBRA (20 hours)

### 1. Introduction

Many applications from engineering and defense require some mathematical backgrounds.
For instance, in cryptography, which has become an important research area of computer science and applied mathematics, a lot of cryptosystems are based on modular arithmetic and number theory.

This course introduces some basic concepts of abstract algebra such as group, ring, and field, among others. The theory is presented and developed with the familiar examples of integers and polynomials so that the motivation is maintained. Meanwhile, the Chinese remainder theorem and the factorization problem appear naturally in this context.

As an applications of the contents provided, the widely used RSA cryptosystem is explained. Hence this course can be thought of as an introduction to the course "Computational Number. Theory and Public Key Cryptography" for which a good knowledge of the previous concepts and results is required.

### 2. Table of contents

Groups
Rings
Fields
Elementary number theory: the quotient rings Z/nZ
Polynomial ring over a field and their quotients
Chinese remainder theorem
Legendre and Jacobi symbols
Boolean functions
Linear recurring sequences (shift registers) over fields

**3. References/sources**
Lindsay N. Childs: A Concrete Introduction to Higher Algebra, Springer.

Ernst S. Selmer: Linear recurrence relations over finite fields,
Department of mathematics University of Bergen, Norway, 1966.

## COMPUTATIONAL NUMBER THEORY AND PUBLIC KEY CRYPTOGRAPHY (20 hours)

### 1. Introduction

Public key cryptography is “everywhere”. It avoids the necessity of individual key exchanges, which would be infeasible in many contexts.
We try to show some old and new systems of public key cryptography. In order to explain what makes these systems work, a well-pondered minimum of algebraic and number-theoretic background will be provided.

Objectives: Become familiar with basic notions of elementary and algorithmic number theory; understand why PKC (Public Key Cryptography) systems work, and what they are used for.

Principal key words: Rings; computing modulo n; exponentials; RSA; hardness of factorization; finite fields; elliptic curves.

**2. Table of contents**

- Basics of complexity theory (running time, Landau's O notation),
- Ring theory: Euclidean algorithm, Chinese Remainder Theorem, Square and Multiply,
- Finite fields: extensions, polynomials, irreducibility,
- The old chestnut: RSA,
- Other classical public key systems,
- Elliptic curves (definition, examples, group law, point counting),
- Cryptography with (hyper) elliptic curves.

**3. References/sources**

Lindsay Childs, A Concrete Introduction to Higher Algebra, Springer UTM
133 [selected chapters]

Henri Cohen, A Course in Computational Algebraic Number Theory, Springer GTM 138
Neal Koblitz, A Course in Number Theory and Cryptography, Springer GTM 114

**DESIGN, SECURITY OF CRYPTOGRAPHIC ALGORITHMS, CRYPTANALYSIS (20 hours)**

**1. Introduction**

Symmetric primitives for cryptography form the core of cryptography from ancient times (Ceasar's cipher and Scytale) to advanced encryption standard (AES) passing through the very famous ENIGMA machine. In the modern times, a lot has been made to take into account for security not only the security of the primitive concerning the key recovery problem, but the environment in which these primitives are implemented field programmable gate array (FPGA), application specifique integrated circuit (ASIC), software, radio frequency identification (RFID).
In the design of a cryptographic system it is now necessary to model the possibilities for an attacker to recover a part of information, or to impersonate the right owner of the communication. Therefore, to primitives have been added modes of operation ensuring security in different attacker models.
Another very important problem concerns, the media on which the primitives are implemented, since a study of the electric consumption can give information on what happens during the implementation.

In this course we present the principle which guides the designers to conceive new symmetric primitives, new modes of operation and review the bank of attacks that exist against the different types of primitives as well as their hardware or software implementations.

Objectives
- Te be aware of the design principles
- To know the modes of operation
- To know how to design a secure implementation of symmetric ciphers and modes of operation

Key words
Symmetric cryptography, Symmetric modes, statistical cryptanalysis,
Side-channel attacks, implementation

## 2. Prerequisites

Linear algebra
Boolean algebra
Logics
Complexity bases
Probability bases
Some experience in computer science

## 3. Table of contents

I Introduction on ciphers
- Vernam cipher (unconditional security)
- Stream ciphers
- Block ciphers
- Modes and authenticated encryption
- Hash functions

II Designs
- Stream ciphers
- Iterated block ciphers
- Feistel and DES
- Substitution permutation network (SPN) and AES
- Sponge and Keccak

III Basic Blocks
- Boolean Functions
- "Maximum Distance Separable" linear layers
- Key schedules

IV Standard Attacks
- Correlation attacks on stream ciphers
- Time Memory Tradeoffs
- Algebraic attacks
- Statistical Cryptanalyses (linear & co, differential & co)
- Collision and pre-image

V Side-Channel attacks and Countermeasures
- Timing (AES T-tables)
- Cache (flush and reload)
- Differential power analysis (DPA)
- Algebraic DPA
- Faults attacks

## 4. References/sources

The Design of Rijndael (Daemen, Rijmen 2002)

The Block Cipher Companion (Knudsen, Robshaw, 2011)

A Classical Introduction to Cryptography + Exercise Book (Vaudenay, Baignères, Junod, Lu, Monnerat 2010)

Power Analysis Attacks (Mangard, Oswald, Popp, 2007)

**PROOFS OF SECURITY (20 hours)**

## 1. Introduction

In modern cryptography, one cannot rely anymore on heuristics to declare that a system in secure. The complexity of both the systems and the attackers forces a systematic and formal approach to the notion of security.

Provable security allows us to understand, to state, and to ensure the security level of cryptographic algorithms with clear assumptions about the adversary's access to the system (i.e. the power of the adversary) and the hardness of some computational tasks (e.g. factoring).

We will show how, from a small subset of hard problems, we can build various and powerful cryptographic tools such as provably secure encryption, secure multi-party computation, or zero-knowledge proofs.

## 2. Prerequisites

Basic notions of probabilities and logic
Complexity and reductions of security

## 3. Table of contents

Objectives
- Understand and write security reductions
- Be able to prove the security of an encryption scheme
- Explain how MPC and ZK-proofs work

Keywords
Computational hardness, security reductions, provable security, secure design of cryptosystems, zero-knowledge

-I) Significance of Security
- Indistinguishability
- Computational / statistical hardness
- Security games

-II) Computational hardness
- One-way functions / One-way permutations
- Trapdoor permutations
- Computational assumptions,
- Reductions

- III) Secure encryption
- Semantic security
- Example of the Goldwasser-Micali encryption scheme
- Improvements of the security definitions

- IV) Zero-knowledge proofs
- Security game
- Bit commitments

- V) Multi/Two party computation
- Definition
- Oblivious transfer
- Garbled circuits

**4. References/sources**

The Foundations of Cryptography de Oded Goldreich
(http://www.wisdom.weizmann.ac.il/~oded/foc-book.html)

Lecture Notes on Cryptography de Goldwasser et Bellare
(http://cs.brown.edu/courses/csci1510/reference/goldwasser_bellare_notes.pdf)

## IV HUMANITIES

## THE ART OF PUBLIC SPEECH (20 hours)

### 1. Introduction

The art of communicating effectively in an international environment whatever the professional context is a vital skill and an important dimension of leadership for officers.

To achieve this aim, a prime requirement lies in the acquisition by the young officers of a good command of both the cultural references typical of our partners in the English-speaking world, and of the various ways in which they usually choose to convey their message.

From the briefing to the motivational speech, from command-type situations to situations of direct conflict, from humour to persuasion, working on a whole range of communication situations will enable the young officers to discover and practice the skills which will enable them to gain in self-confidence and effectiveness, hopefully enjoying themselves in the process.

Based on extracts from original films covering a cross section of history from the times of King Henry V to World War II, Vietnam, ex-Yugoslavia, Rwanda or more recent peace time situations, after a presentation of the excerpts in context, the classes will be as interactive as possible, also allowing the cadets to debate the situations concerned or even to suggest their own material.

The young officers will be required to act out a number of scenarios directly based on excerpts from classic films. After discovering in class the context and the relevant information necessary to fully understand the scenes, they will be called upon to play the scenes themselves, eventually learning by heart certain excerpts and presenting them in front of the group. Their performance will be filmed (at the Military Academy's CMAF), discussed collectively and ultimately assessed by the professor in charge of the course. 20 percent of the marks will be awarded for overall participation, 60 percent for individual performance, with a final 20 percent written assessment of progress in the command of cultural and historical knowledge, and the nuances of speech (intonation, understatements, etc).

### 2. Table of contents

- **Henry V** (Laurence Olivier / Kenneth Branagh): motivational speech before the battles of Harfleur and Azincourt (We band of brothers ...) –to be compared with LtCol Tim Collins's Iraq 2003 speech

- **Waterloo**, by Dino de Laurentiis (1970), with Rod Steiger, Christopher Plummer and Orson Welles: Napoleon and Wellington, leadership styles and command situations

- **Glory** (1989): American Civil War difficult leadership situations for a young commanding officer

- **Paths of Glory,** Stanley Kubrick (1957): defending subordinates.

- **The King's speech** (2010): mastering the art of public delivery.

- **Patton** (1970): motivational speech; relations between Patton and Gen. Omar Bradley.

- **Gardens of Stone** (Francis Ford Coppola, 1985): relations between officers and NCOs in a Vietnam War environment.

- **Warriors** (1999): giving a briefing ‘in-theater’; coping with very tense situations; conducting a difficult negotiation.

- **Crimson Tide** (2003): a modern-day ‘Caine Mutiny’ type scenario on board an SSBN.

- **Hotel Rwanda** (2004): a very tense confrontation with a Rwandan general.

- **An Inconvenient Truth** (2006): the ways in which a skillful politician chooses to communicate.

Original course material will be provided to the cadets either in advance of every class or ‘on the spot’ –and made available online on the Academy’s internet site.

## FINANCIAL CRISES AND VIOLENCE (20 hours)

### 1. Introduction

The course focuses on the relations between financial crises and violence. It focuses on a current geopolitical conflict. It examines the historical, geopolitical, economical and cultural reasons of the war. The class is divided into several groups in order to carry out research on these different aspects.

#### Conclusions of the seminar carried out in 2016-2017

These conclusions give an example of the work carried out by the international cadets:

Already foreseen in 2004 by the National Intelligence Council, the advent of Daesh is not a surprise for everyone. Its nature is twofold, since the organization presents itself both as a political construction and a dream of taking revenge on the armies, which defeated the Caliphate in the thirteenth century. Daesh is a religious response to the Baathist secularism as well as to Shiia pietism. The Islamic state remains however a fragile construction, based on tribal opportunism. Its leaders, who present themselves as degraded officers, use the organization to pursue their own end: that of regaining power. In these circumstances, how can we explain the expansion of the Islamic state? The economic disruption it causes clearly benefits the Saudi interests while harming China, which covets the Iraqi oil fields. This state attracts jihadists insofar as it presents itself as the answer to a humiliation. Since the thirteenth century, the Arab world has indeed lost control of its political destiny. Daesh, which is at the same time a rational and totalitarian organisation, has won the esteem of the people by struggling against chaos in the conquered territories. Well-armed, endowed with substantial funds, it is adapted to the coming world: Al Qaeda was soluble in globalization, but the Islamic state has had the intuition that the world of tomorrow would be composed of nations. Reconnecting with the past, it has taken a step ahead. To defeat it, the Western coalition will have no choice but to consider a permanent political solution. Otherwise, the war will go on.

### 2. References/sources

Carmel S., “Commercial shipping and the maritime strategy, *Naval war college review*, spring 2008, pp. 40-46

Evans M., “Lethal Genes: the urban military imperative and western strategy in the early 21st century”, *The Journal of Strategic studies*, vol. 32, N° 4 pp. 515-552

Ghosh, B., “The global financial and economic crisis and migration governance”, *Global governance,* 16, 2010, pp. 317-321.

Gills, B. K.,”Going South, capitalist crisis, systemic crisis, civilizational crisis”, *Third World Quaterly,* vol. 31 n° 2, 2010, pp.169-184.

High H., “ Laos, crisis and resource contestation”, *Southeast Asian Affairs,* 2010, pp. 153-161.

Howard, R., “Peak oil and strategic wars”, *The futurist,* September-october 2009, pp. 18-21.

Lind M., “A concert-balance strategy for a multipolar world”, *Parameters,* autumn 2008, pp. 48-61

Mac Michael P., “Contemporary contradictions of the global development project: geopolitics, global ecology and the development climate”, *Third World Quaterly,* vol. 30, n° 1, 2009, pp. 247-262

Voeller, J. G., “The era of insufficient plenty”, *Mechanical engineering,* June 2010, pp. 35-39.