INFORMATION TECHNOLOGY
ИНФОРМАТИКА
School of Advanced Studies
Quarter 4, April 13 to June 11, 2020

Instructor/s: Fabio Grazioso f.grazioso@utmn.ru
Vitaly Nikolaev v.nikolaev@utmn.ru

Available for consultation via pre-scheduled Zoom appointment

Course Coordinator: Fabio Grazioso f.grazioso@utmn.ru

Contact Hours: 64

Type of Course: Core course for the first-year students

Meeting Times:
group 1 (VN): **Tuesday**: 14:20-15:50, 17:40-19:10, **Thursday**: 16:00-17:30, 17:40-19:10
group 2 (FG): **Tuesday**: 14:20-15:50, **Thursday**: 16:00-17:30, 17:40-19:10, **Friday**:12:30 - 14:00
group 3 (FG): **Monday**: 17:40-19:10, **Tuesday,** 17:40-19:10, **Wednesday**: 17:40-19:10, **Friday**:17:40-19:10

Course Description

This IT course is designed to give practical knowledge, and considers mostly “the computer as a tool”. The course is designed with the assumption that students already have some working knowledge of computers and the internet. The course will still begin with some brief theoretical general definitions about computers, hardware, and software. It will then focus on some advanced use of software tools for word processing, spreadsheets, presentations, scientific typesetting, and graphic editing. We will then examine theoretical notions for sound code design, illustrating different coding approaches. It will then give knowledge on programming, with focus on applications and concrete examples. The programming will be done mostly in Python language, with some examples of C language. The programming will be done in practical sessions.
Due to COVID-19 contingency, the course will be delivered online via Zoom conferencing software.

Course Structure

The course will develop over 8 weeks, with 32 classes of 90 minutes. 20 classes will take the form of lectures, and 12 will be practical sessions.

Based on the results of the preliminary tests, all the students have been divided into three groups - basic, intermediate and advanced. The three groups will study the same topics but with a different level of discussion.

Student Learning Goals

Students who successfully pass this course will be able to:

Learning goals	
Knowledge goal:	Understand basic concepts and technical definitions of computer science and computer networks.
Knowledge goal:	Understand and implement some programming techniques, with emphasis on applications in the social sciences and humanities.
Practical skill:	Program in the Python language, and will be able to understand the theory behind it, as much as to write code for the solution of different problems.
Practical skill:	Understand and troubleshoot some basic problems of computer hardware.

Required Coursework and Evaluation Criteria

The final grade for this course will be calculated as follows:

Assignment or Task	Due date/s	Percent
Participation	Ongoing	40
Intermediate test	end of the first half of the course (4th week)	30
Final test	end of the course (8th week)	30

This course employs 7-median (the number of grades above 7 and the number of grades below 7 do not differ by more than 1)Failing grades (0-3) are not included in the calculation of the 7-rule. The 7-rule will be applied across all sections in the entire course.

All marks are provisional until the end of the course. The 7-rule WILL NOT be used in assessing individual assignments. It will only be applied to the final course marks, pending overall student performance. If general performance is low, a lower overall median/average may apply – if performance is outstanding, a higher overall median may apply.

Intermediate Test (30 %)

An intermediate test will be given halfway through the course. The test will be conducted with the use of an online form, with questions and problems regarding the material seen up to that point in the course.
The performance will not receive a quantitative evaluation, but the quality of the performance will be discussed individually with the students.
The evaluation criterion will be whether the answers to the several questions of the test are satisfactory or not. Some questions have a list of possible answers, some questions have open answers.
Here are three sample questions:

1) Describe the "Turing Machine" and its main parts.
2) Give a brief description of the Operating System. What is its purpose? What are its main tasks?
3) What is the meaning of the acronym "wysiwyg"?

Final Test (30%)

The final test will be given at the end of the course. The test will be conducted in the online class and will consist mostly of programming tasks in Python language. The performance will not receive a quantitative evaluation.
Example questions:

1) Ask the user for his/her name, then print a greetings message containing the name, something like “Good evening ***, how are you?”. (*)
2) Ask a user for a number n and print the sum of the numbers from 1 to n. (*)
3) Using *regular expressions*, search within a long string variable and find all the *credit card numbers* (16 digits), including the case where there are spaces between groups of 4 digits. In other words search for #### #### #### #### or for ################.

Optional: with a *text editor* save the long text containing credit card numbers in a file. Then open the file from python, and load the text into a string variable. Then apply the regular expression search to the string variable.

Participation (40%)

Participation will be based on the behaviour of the student throughout the whole course, taking into account the attendance to the lectures and computer practices, participation in discussions, and the effort put in the interactive computer practices.
General interaction with the teacher during the whole course: informal questions and interaction during the theoretical lectures, and one-to-one work with each student during the practice.

Hardware requirements for the course

A computer with internet connection would be required.
A non-ideal situation would be attending through a smartphone: still doable, but with much poorer results.

Canvas and Other Course Resources

This course has a website on Canvas (https://canvas.instructure.com/). You should have received an invitation to join the course on Canvas two weeks before the start of classes. If you did not, double check your SAS email and then follow up with the instructor. All course readings, this syllabus, and any other course materials are available on Canvas.

All written assignments completed outside of class must be submitted via Canvas.

Course Literature

Here is a bibliography of literature that will assist you in studying and writing assignments. See the course schedule below for specific reading assignments.

1) Fox - Information Technology: An Introduction for Today's Digital World (2013).
2) Vermaat, Sebok, Freund - Discovering Computers 2017 (2016).
3) Tanenbaum, Bos - Modern Operating Systems (2014).
4) Harris, Harris - Digital Design and Computer Architecture (2nd Ed) (2012).
5) Mueller - Windows command line administration instant reference (2010).
6) Gillam - Unicode Demystified (2002).
7) Hunt - TCP-IP network administration (2002).
8) Martin - Clean Code- A Handbook of Agile Software Craftsmanship (2008).
9) Cormen - Algorithms unlocked (2013).
10) Sauer - Numerical Analysis (2011).
11) Atkinson - An Introduction to Numerical Analysis (1989).
12) Burden, Faires, Burden - Numerical Analysis (2016).
13) Papoulis, Pillai - Probability, Random Variables and Stochastic Processes (2002).
14) Jaynes - Probability Theory The Logic of Science (2003).
15) Lutz - Learning Python (2013).
16) Langtangen - A primer on scientific programming with Python (2012).
17) Friedl - Mastering Regular Expressions (2006).
18) Baecker - Readings in human-computer interaction (1995).
19) Norman - The Design of Everyday Things (2002).

Course Policies and Expectations

Students are expected to interact with the teacher, ask questions, suggest examples, and engage in discussions about the subjects presented in online class.

Programming practice is required as part of the course, and the teacher will interact as much as possible individually with each student.

Examination Format

The examination consists of a 45-minute test that consists of 20 questions. For full details on the format and grading, see the SAS policies section below.

Course Schedule

During lectures there will be some interaction anyways (asking informal questions to the students, involving them in discussing the topics) but it will be limited. The practice will be highly interactive, with the professor assigning tasks and problems to solve, and then spending time with each student individually.

Week	Date	Activity	Topics & Readings	Assignments
0 (13-19 Apr)	G.1: 14.04 G.2: 14.04 G.3: 13.04	Lecture	General architecture of a computer. The Von Neumann machine. The machine representation of numbers and errors (logical). The binary numbers, the hexadecimal numbers. Encodings. The operating system in general. MS-DOS, Graphical Operating Systems: MS Windows MacOS.	none
	G.1: 14.04 G.2: 16.04 G.3: 14.04	Lecture	Word processors. Spreadsheets. Presentation. Graphic editors. Audio editors. Text editors. Utility software, examples.	
	G.1: 18.04 G.2: 16.04 G.3: 15.04	Practice session	Computer Lab practice on word processing, spreadsheets, presentation software.	
	G.1: 18.04 G.2: 17.04 G.3: 17.04	Practice session	Computer Lab practice on graphic editing (both vector and raster) for academic applications.	
1	G.1: 21.04	Practice session	Computer Lab practice on audio editing and Scientific Typesetting (LaTeX)	none

(20-26 Apr)	G.2: 21.04 G.3: 20.04			
	G.1: 21.04 G.2: 23.04 G.3: 21.04	Lecture	Networking protocols, layers, packets. TCP/IP protocol. World Wide Web, HTML, browsers. Search engines. Other protocols: email, FTP, servers, clients.	
	G.1: 23.04 G.2:223.04 G.3: 22.04	Practice session	Computer Lab practice on internet resources: academic resources, advanced search engines practice.	
	G.1: 23.04 G.2: 24.04 G.3: 24.04	Lecture	Main programming language types: machine code, low-level, high-level, interpreted, compiled. Syntax and semantics.	
2 (27 Apr - 3 May)	G.1.: 28.04 G.2: 28.04 G.3: 27.04	Lecture	Programming environments: editor, debugger, compiler, interpreter. Software design techniques. Algorithms.	none
	G.1:28 .04 G.2: 30.04	Lecture	The Python language. Keywords and syntax. Control flow statements. Data types.	

	G.3: 28.04			
	G.1: 30.04 G.2: 30.04 G.3: 29.04	Practice session	Computer Lab practice on Python programming: first simple algorithms.	
	G.1: 30.04 G.2: 1.05 G.3: 1.05	Lecture	Python Libraries: NumPy, SciPy. Examples of mathematical functions, mathematical data generation and manipulation. Examples of simple scientific plots.	
3			no classes this week	
4 (11-17 May)	G.1: 12.05 G.2: 12.05 G.3: 11.05	Practice session	Computer Lab practice on Python programming: first scientific applications.	mid course test
	G.1: 12.05 G.2: 14.05 G.3: 12.05	Practice session	Computer Lab practice on Python programming: more scientific applications.	
	G.1: 14.05 G.2: 14.05	Lecture	Python data processing. Examples of data plotting and representation: 2D plots, 3D plots, annotations.	

	G.3: 13.05			
	G1.:14 .05 G.2: 15.05 G.3: 15.05	Test	Mid-course test	
5 (18-24 May)	G.1: 19.05 G.2: 19.05 G.3: 18.05	Practice session	Computer Lab practice on Python programming: scientific data plotting.	none
	G.1: 19.05 G.2: 21.05 G.3: 19.05	Lecture	Python datafiles input and output. Text files and binary files. Strings Manipulation. Regular Expressions.	
	G.1: 21.05 G.2: 21.05 G.3: 20.05	Practice session	Computer Lab practice on Python programming: data files input and output.	
	G.1: 21.05 G.2: 22.05 G.3: 22.05	Lecture	Numerical analysis. Direct methods and iterative methods. Precision. Example algorithms: sorting, searching.	

6 (25-31 May)	-	Lecture asynch.	More numerical analysis. Root-finding, linear equation systems solving, functions interpolation, functions integration. Discrete Mathematics.	
	-	Practice asynch.	Computer Lab practice on Python programming: applications on Numerical analysis and discrete mathematics.	
	-	Practice asynch.	Computer Lab practice on Python programming: applications on Numerical analysis and discrete mathematics.	
	-	Lecture asynch.	Statistics and Probability. Fundamentals of statistics. Fundamentals of probability theory. Bayes theorem.	
7 (1-7 June)	-	Practice asynch.	Computer Lab practice on Python programming: applications on statistics.	
	-	Practice asynch.	Computer Lab practice on Python programming: applications on statistics.	
	-	Practice asynch.	Computer Lab practice on Python programming: Examples from the course.	
	-	Practice asynch.	Computer Lab practice on Python programming: Examples from the course.	
8 (8-13 June)	-	Practice asynch.	Computer Lab practice on Python programming: Examples from the course.	
	-	Practice asynch.	Computer Lab practice on Python programming: Examples from the course.	
	-	Practice asynch.	Computer Lab practice on Python programming: Examples from the course.	

	G.1: 11.06 G.2: 11.06 G.3: 10.06	asynch.	A final test	

SAS Policies for Online Courses

Please note the addition and updating of policies to reflect the realities of online teaching in Q4.

Technical Requirements and Responsibilities for Online Education

Professors and students are responsible for ensuring they have access to a computer and a stable Internet connection during all scheduled class meetings. This is to ensure that students get the most out of the online education format. If you have problems with your Internet, smartphones may be used as a backup option (as a wifi hotspot or to participate in class).

Course materials and all assignments will be made available on Canvas; all synchronous class meetings will be conducted over Zoom. All communication about the course and assignments must happen over Canvas or official email. The use of any supplementary platforms (discussion boards etc.) is at the discretion of the instructor.

Professors are required to post all resources for online teaching via Canvas before the start of each week. This includes: Any nonsynchronous lesson material, the invitations for individual Zoom meetings, and any other materials required to complete the course.

All synchronous classes will be recorded and made available via Canvas on the same day for a minimum of one week. These recordings are only for teaching purposes and should not be shared.

Etiquette for Online Classes

Professors and students should join Zoom a few minutes before class in order to have time to solve any technical problems. When you join a class, your microphone will be muted. Individual professors will decide how to run class discussions and whether to enable such features as chat. As a general rule of thumb, you should mute your microphone when you are not speaking.

In seminars, students are required to make themselves visible. If you have concerns about what is visible, then either take the time to "curate" your environment or consider using the background option in Zoom. During lectures, you are welcome to turn off your video.

Students should feel free to contact the professor or Head of Education (d.kontowski@utmn.ru) to discuss any concerns that may arise concerning online delivery of the course (i.e., technical issues, course material availability, access to apps, communication challenges, and changes to syllabus or schedule). Don't wait until course evaluations to draw attention to your concerns!

Technical Emergencies Protocols

Students who have difficulty getting online to attend a synchronous class or complete an assignment, should contact the professor immediately according to the specific instructions provided in the syllabus (i.e., via telephone, SMS, or email). Follow the below instructions concerning making up classes missed due to technical problems.

If your professor is not online for the start of a class session, keep Zoom open and check your email. If the professor does not come on-line or send a message to clarify the situation within 10 minutes after the official starting time, class is cancelled. Both the professor and a designated student should alert the Head of Education about the situation. Missed classes will be rescheduled; update class times to be shared via Canvas and Modeus.

Attendance and Absences

Zoom has an attendance feature that will be used to record attendance. Attendance is required for all synchronous classes or required online activities (i.e., designated asynchronous tasks, timed assignments, group work meetings, etc.) and will be recorded on a grading sheet. Students can miss up to two classes without an excuse; every further absence will see the final mark lowered by 1 point for each class missed (i.e., a student who misses 6 class meetings without prior approval or a valid excuse cannot pass a course). Missing more than 15 minutes of scheduled online class is considered an absence, unless the student has received prior approval from the Head of Education.

If you plan to miss a class due to a legitimate conflict (i.e. attendance of a student conference), you must apply to the instructor for an approved absence at least seven days in advance and CC Head of Education. Without advanced approval, it will count as a missed class.

If you are sick, email all your instructors and Alyona Bunkova (a.bunkova@utmn.ru) as soon as possible to notify them that you will be missing class. They will follow up with you with any necessary arrangements related to your illness.

If you need to miss a class due to something that arises at short notice (i.e., bureaucracy that needs to be dealt with, an emergency at home), email the instructor as soon as possible to notify them about your absence. Should a student have repeated problems with attendance, the instructor will notify the Head of Education.

Making Up Classes Missed for Legitimate Reasons

Students who miss a synchronous class session to a legitimate conflict, an emergency that arises at short notice, or a technical problem will be required to watch the recording of the class and submit a written summary of the key points of the class, including any questions that you have about the content. This should be sent to the instructor via email within 48 hours of the ending of the class in order to receive credit. If a technical problem emergency situation persists beyond 48 hours, an extension may be granted. Students who are sick should watch the videos of missed classes in order to keep up on courses, but they are not required to submit written summaries.

Extensions for Assignments

All assignments must be submitted by their due dates. Extensions will be granted only when ill health, death of a loved one, or personal difficulties of a serious nature near the due date prevent completion of an assignment. As the due dates for assignments are stated in the syllabus, the pressure of other university work or extracurricular activities will not be accepted as a reason for an extension.

If you require an extension, you must write to your instructor at least three working days in advance. Clearly explain your situation and provide any necessary documentation (such as a medical certificate) to Alyona Bunkova. Your instructor should reply to you within one day; you will be notified by email about whether an extension has been granted.

Late Assignments

Late assignments will be penalized by a full grade deduction for each day of lateness. For example, an essay submitted three days late that received a mark of 7 would be reduced to 4. Late assignments will not be accepted once graded assignments are returned or after June 11. The acceptance of late assignments for minor assessments (worth 10 percent or less of the final mark, including minor tasks completed during class hours) is left up to the discretion of individual instructors.

Rescheduling of Classes or Substitution of Instructor

Should a course be unable to meet at its regular time, the instructor will liaise with Alyona Bunkova to approve the change and to find a different time that suits both the instructor and students. Should this occur, all involved will receive an email notification from Alyona Bunkova about the changed schedule and any schedule changes will appear in Modeus. If the instructor requires a substitute to replace them, students will be notified by email.

Grading

SAS uses a ten-point grading system. Grades from 0 to 3 are failing grades. Grades from 4 to 10 are passing grades. 10 and 9 are excellent grades given in exceptional circumstances.

In most courses, SAS faculty are obliged to follow the 7-rule. This may be calculated either as a “median” (the number of grades above 7 and the number of grades below 7 do not differ by more than 1) or an average (the average final grade for all students should fall between 6.50 and 7.49). The 7-rule may be applied to each assignment OR only to the final course marks. Exceptions to this rule are only granted by the Teaching Council.

Examinations

The examination will consist of 20 questions and lasts for 45 minutes (5 minutes to prepare, 40 minutes to write answers). The use of any electronic devices is prohibited. The examination may include the following types of questions: 1. Multiple choice questions; 2. Fill in the blanks questions; 3. Give the output for the given program (snippet). The number of questions of each type may vary, but

there will be 20 questions total. All answers must be written in English or in the relevant program code.

Assessment Criteria:

Satisfactory (C, or 3):	80% to 87% correct answers.
Good (B, or 4):	88% to 95% correct answers.
Excellent (A, or 5):	96% to 100% correct answers.

Course Evaluations

Toward the end of the quarter, students will be asked to complete an anonymous evaluation of the course. The results of the evaluations will be reviewed by the instructor, the Head of the Education Office, and the Teaching Council in order to improve education at SAS.

Academic Integrity

Students are expected to comply with the SAS Academic Integrity Document (see English version HERE or Russian version HERE). Cheating, plagiarism, and disrespectful behavior will not be tolerated and *must* be sanctioned by the instructor in accordance with the document. The use of any translation applications (Google Translate etc.) is highly discouraged. Students are required to cite any sources employed in written assignments using the citation style listed in the syllabus.

Online assignments will be “open book,” meaning that you can look at course reading materials and notes while answering the questions. However, the Academic Integrity still applies. That means: You must not communicate with anyone; your answers will be your own work; and you will not use Google Translate. You are discouraged from searching the Internet for answers, as you will run out of time, may risk violation of the Academic Integrity Policy, and will likely do worse than if you simply answer with the knowledge you already have.

Date Syllabus Last Updated: 16.04.2020