**KAZAKH-BRITISH TECHNICAL UNIVERSITY**

**SCHOOL OF APPLIED MATHEMATICS**

**Approved by**

**Acting dean of School of Applied**

**Mathematics**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_T.S.Kenzhebayev**

**Syllabus**

**Programming on Python II MAT 1325**

Semester: Fall 2024

2024/2025 Academic Year

3 credits (1/0/2)

**Instructor:** Aizhan Bolatovna Abylkassymova, associated professor

| **Personal Information about the Instructor** | **Time and place of classes** | | **Contact information** | |
| --- | --- | --- | --- | --- |
| **Lessons** | **Office Hours** | **Tel.:** | **e-mail** |
| Senior lecturer | According to the schedule | TBA | +7 707 8034921 |  |
|  |  | TBA |  | [abylkassymova.aizhan@gmail.com](mailto:abylkassymova.aizhan@gmail.com) |

**COURSE DURATION:** 3 credits, 15 weeks, 45 class hours

**COURSE PREREQUISITES**: Basic Python programming experience. In particular, you should be very

comfortable with: working with strings, working with lists, tuples and dictionaries, loops and conditionals, writing your own functions.

**COURSE DESCRIPTION**

# Course Objectives:

The course is aimed for highly motivated students with basic knowledge of programming in Python. The course will focus on planning and organizing programs, as well as the grammar of the Python programming language for mathematical and computer modeling problems.

This course is a great opportunity to both improve and take out a lot of useful libraries and tips on programming concepts. By the end, you'll be familiar with advanced Python syntax and you'll be able to put into practice what you'll have learned in a final project you'll develop locally. Students will understand the methodology for programming with advanced Python based on mathematical and computer modeling problems.

# Competition (learning outcomes):

After attending Programming on Python II course, students will be able to perform

the following tasks:

 Design, develop, and implement powerful unit testing within their Python applications.

 Understand usage of main libraries: math, random, GUI library, numPy, matplotlib, pandas and others.

 Understand and leverage Object Oriented programming techniques in their Python applications.

 Alter or augment the operation of existing or inherited Python code using decorators

 Apply powerful regular expression matching and manipulation techniques.

 Utilize sockets; networking, including Python provided protocol clients

 Leverage and understand Python's threading and multiprocessing module, including locks, conditions, reentrant locks, and a host of other related topics.

 Develop Python applications that react to signals.

 Understand, develop, and deploy cooperative multitasking

**In evaluating the performance of the student during the semester to consider the following:**

* Attendance
* Active and productive participation in practical exercises
* Study of basic and additional literature
* Homework
* Implementation of the projects
* Timely delivery of all jobs

**REFERENCES**

**Main:**

**Basic Literature:**

1. John M. Zelle. Python Programming: An Introduction to Computer Science. 2002.

2. Alex Martelli. Python in a Nutshell: A Desktop Quick Reference. 2003.

# 3. Vernon L. Ceder. The Quick Python Book. 2010.

**Supplementary literature:**

1. Brian Heinold. A Practical Introduction to Python Programming. Department of Mathematics and Computer Science Mount St. Mary’s University. 2012.

#### COURSE CALENDAR

| **Week** | **Class work** | | | | **SIW (student’s individual work)** |
| --- | --- | --- | --- | --- | --- |
| **Topic** | **Lectures** | **Seminars** | **Chapters for reading** |
| 1 | What is computation?  Overview to Python programming, terminology and main concepts, Basic Syntax, variable and Data Types. Operators  *Seminar #1.* | 1 | 2 | according to the lecture notes |  |
| 2 | Quick overview of basic Python syntax (Loops, conditions, data types, functions, tuples, files, dict)  *Seminar #2.* | 1 | 2 | according to the lecture notes |  |
| 3 | Input-Output. Printing on screen.  Reading data from keyboard.  Opening and closing file.  Reading and writing files Functions  *Seminar #3.* | 1 | 2 | according to the lecture notes |  |
| 4 | Regular Expressions  *Seminar #4.* | 1 | 2 | according to the lecture notes |  |
| 5 | OOPs concept  Class and object  Inheritance  Overloading  Overriding  Data hiding  *Seminar #5.* | 1 | 2 | according to the lecture notes |  |
| 6 | Databases, Math (cmath), Random  *Seminar #6.* | 1 | 2 | according to the lecture notes |  |
| 7 | GUI Programming  Tkinter programming  Tkinter widgets  Interface design  *Seminar #7.* | 1 | 2 | according to the lecture notes |  |
| 8 | NumPy, SciPy  *Seminar #8.* | 1 | 2 | according to the lecture notes |  |
| 9 | Pandas  *Seminar #9.* | 1 | 2 | according to the lecture notes |  |
| 10 | MatPlotLib, Seaborn  *Seminar #10.* | 1 | 2 | according to the lecture notes |  |
| 11 | *Pandas +NumPy+MatPlotlib*  *Seminar #11.* | 1 | 2 | according to the lecture notes |  |
| 12 | What is Machine Learning?  Introduction to ML, terminology and main concepts. Supervised, Unsupervised learning introduction.  *Seminar #12.* | 1 | 2 | according to the lecture notes |  |
| 13 | Model Representation. Cost function, Intuition 1, Intuition 2.  Gradient Descent, Gradient Descent Intuition, Gradient descent for Linear Regression.  *Seminar #2. Review for Python, Overview Octave/Matlab*  *Seminar #13.* | 1 | 1 | according to the lecture notes |  |
| 14 | Multivariate Linear Regression. Computing Parameters Analytically  *Seminar #3. Programming Assignment #1 Linear Regression*  *Seminar #14.* | 1 | 2 | according to the lecture notes |  |
| 15 | Classification and Representation: Classification, Hypothesis Representation, Decision Boundary. Logical Regression Model. (Cost function, Simplified Cost Function and Gradient Descent, Advanced Optimization )  Multiclass Classification. One vs. all. Regularization (The problem of Overfitting, Cost Function, Regularized Linear/Logistic Regression)  *Seminar #4. Programming Assignment: Logistic Regression*  *Seminar #15.* | 1 | 2 | according to the lecture notes |  |

**COURSE ASSESSMENT PARAMETERS**

| Attendance and activity on lessons | 12% |
| --- | --- |
| Quiz/Projects | 33% |
| Home works | 15% |
| Final exam | 40% |
| **Total** | **100%** |

| **No** | **Assessment criteria** | **Weeks** | | | | | | | | | | | | | | | | **Total** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** |  |
| 1. | Attendance and activity on lessons | 1 | 1 | 1 | 1 | 1 | 1 |  | 1 | 1 | 1 | 1 | 1 | 1 |  |  |  | 12 |
| 2. | Ouiz/Projects |  |  | 5 |  |  |  |  |  |  |  |  |  |  |  |  |  | 5 |
| 3. | Activity/Home works | 1 | 1 | 1 | 1 | 1 | 1 |  | 4 | 1 | 1 | 1 | 1 | 1 |  |  |  | 15 |
| 4 | Projects |  |  |  |  |  |  | 13 |  |  |  |  |  |  | 15 |  |  | 28 |
| 5. | Final  examination |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 40 | 40 |
|  | Total |  |  |  |  |  |  | 30 |  |  |  |  |  |  |  | 30 | 40 | 100 |

**Lectures** are conducted in the form ofexplainingthe theory given in the course that is why students are supplied with handouts uploaded into the intranet. Activity and attendance in lessons is mandatory. Mandatory requirement is preparation for each lesson.

**Grading policy:**

Intermediate attestations (on 7th and 15th week Proctoring System could be used ) join topics of all lectures, laboratories, homework, quiz and materials for reading discussed to the time of attestation. Maximum number of points within attendance, activity, homework, quiz/projects and laboratories for each attestation is 30 points.

Final exam will be a team project. Final exam duration is 120 min. Maximum number of points is 40. At the end of the semester you receive overall total grade (summarized index of your work during semester) according to conventional KBTU grade scale.

### ACADEMIC POLICY

**Students are required:**

* to be respectful to the teacher and other students;
* to switch off mobile phones during classes;
* DO NOT cheat. Plagiarized papers shall be graded with zero points!
* to come to classes prepared and actively participate in classroom work; to meet the deadlines;
* to enter the room before the teacher starts the lesson;
* to attend all classes. No make-up tests or quiz are allowed unless there is a valid reason for missing it;
* to follow KBTU academic policy regarding **W, AW, I, F** grades.
* When students are absent for 20% of the lessons or more (without Spravka), then their grade is F.
* When students have a score of 29 or less for attestation 1 added to attestation 2, then their grade is F.
* When students have a score of 19 or less (less than 50%) for their final exam, then their grade is F.
* When students do not come for their final exam, then their grade is F.

**Students are encouraged to**

* consult the teacher on any issues related to the course;
* make up within a week’s time for the works undone for a valid reason without any grade deductions;

Associated Professor of School of Applied Mathematics Abylkassymova A.B.

Considered in the meeting of School of Applied Mathematics, minutes # «1» 28.08.2024