**ANNA UNIVERSITY, CHENNAI**

**AFFILIATED INSTITUTIONS**

**B.E. COMPUTER SCIENCE AND ENGINEERING**

**REGULATIONS – 2017**

**CHOICE BASED CREDIT SYSTEM**

**PROGRAM EDUCATIONAL OBJECTIVES (PEOs):**

1. To enable graduates to pursue higher education and research, or have a successful career in industries associated with Computer Science and Engineering, or as entrepreneurs.
2. To ensure that graduates will have the ability and attitude to adapt to emerging technological changes.

**PROGRAM OUTCOMES (POS):**

Engineering Graduates will be able to:

1. **Engineering knowledge**: Apply the knowledge of mathematics, science, engineeringfundamentals and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis**: Identify, formulate, review research literature, and analyze complexengineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions**: Design solutions for complex engineering problems anddesign system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems**: Use research-based knowledge andresearch methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage**: Create, select, and apply appropriate techniques, resources, and modernengineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society**: Apply reasoning informed by the contextual knowledge to assesssocietal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and sustainability**: Understand the impact of the professional engineeringsolutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics**: Apply ethical principles and commit to professional ethics and responsibilities andnorms of the engineering practice.
9. **Individual and team work**: Function effectively as an individual, and as a member or leaderin diverse teams, and in multidisciplinary settings.
10. **Communication**: Communicate effectively on complex engineering activities with theengineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

1

1. **Project management and finance**: Demonstrate knowledge and understanding of theengineering and management principles and apply these to one‘s own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
2. **Life-long learning**: Recognize the need for, and have the preparation and ability to engagein independent and life-long learning in the broadest context of technological change.

**PROGRAM SPECIFIC OBJECTIVES (PSOs)**

To analyze, design and develop computing solutions by applying foundational concepts of Computer Science and Engineering.

To apply software engineering principles and practices for developing quality software for scientific And business applications.

To adapt to emerging Information and Communication Technologies (ICT) to innovate ideas and solutions to existing/novel problems.

Mapping of POs/PSOs to PEOs

Contribution 1: Reasonable 2:Significant 3:Strong

2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **PEOs** | |  |  |
|  |  |  |  |  |
| **POs** | 1. Graduates will pursue | 2. | Graduates will |  |
|  | higher education and |  | have the ability |  |
|  | research, or have a |  | and attitude to |  |
|  | successful career in |  | adapt to |  |
|  | industries associated |  | emerging |  |
|  | with Computer |  | technological |  |
|  | Science and |  | changes. |  |
|  | Engineering, or as |  |  |  |
|  | entrepreneurs. |  |  |  |
|  |  |  |  |  |
| 1. **Engineering knowledge**: Apply the |  |  |  |  |
| knowledge of mathematics, science, |  |  |  |  |
| engineering fundamentals, and an | 3 |  | 1 |  |
| engineering specialization to the solution of |  |  |  |  |
| complex engineering problems. |  |  |  |  |
| 2. **Problem analysis**: Identify, formulate, |  |  |  |  |
| review research literature, and analyze |  |  |  |  |
| complex engineering problems reaching | 3 |  | 1 |  |
| substantiated conclusions using first |  |  |
|  |  |  |  |
| principles of mathematics, natural sciences, |  |  |  |  |
| and engineering sciences. |  |  |  |  |
| 3. **Design/development of solutions**: Design |  |  |  |  |
| solutions for complex engineering problems |  |  |  |  |
| and design system components or |  |  |  |  |
| processes that meet the specified needs | 3 |  | 2 |  |
| with appropriate consideration for the public |  |  |
|  |  |  |  |
| health and safety, and the cultural, societal, |  |  |  |  |
| and environmental considerations. |  |  |  |  |
|  |  |  |  |  |
| 4. **Conduct investigations of complex** |  |  |  |  |
| **problems**: Use research-based knowledge |  |  |  |  |
| and research methods including design of |  |  |  |  |
| experiments, analysis and interpretation of | 3 |  | 2 |  |
| data, and synthesis of the information to |  |  |  |  |
| provide valid conclusions. |  |  |  |  |
|  |  |  |  |  |
| 5. **Modern tool usage**: Create, select, and |  |  |  |  |
| apply appropriate techniques, resources, |  |  |  |  |
| and modern engineering and IT tools |  |  |  |  |
| including prediction and modeling to | 2 |  | 3 |  |
| complex engineering activities with an |  |  |  |  |
| understanding of the limitations. |  |  |  |  |
|  |  |  |  |  |
| 6. **The engineer and society**: Apply reasoning |  |  |  |  |
| informed by the contextual knowledge to |  |  |  |  |
| assess societal, health, safety, legal and |  |  |  |  |
| cultural issues and the consequent | 2 |  | 2 |  |
| responsibilities relevant to the professional |  |  |
|  |  |  |  |
| engineering practice. |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

3

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 7. | **Environment and sustainability**: | | | | |  |  |  |  |  |  |
|  | Understand the impact of the professional | | | | |  |  |  |  |  |  |
|  | engineering solutions in societal and | | | | |  |  |  |  |  |  |
|  | environmental | | contexts, | and demonstrate | | 2 |  | 1 |  |  |  |
|  | the knowledge of, and need for sustainable | | | | |  |  |  |  |  |  |
|  | development. | |  |  |  |  |  |  |  |  |  |
|  |  | | | | |  |  |  |  |  |  |
| **8.** | **Ethics**: Apply ethical principles and commit | | | | |  |  |  |  |  |  |
|  | to professional ethics and responsibilities | | | | | 3 |  | 1 |  |  |  |
|  | and norms of the engineering practice. | | | | |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  | | | | |  |  |  |  |  |  |
| **9.** | **Individual and team work**: Function | | | | |  |  |  |  |  |  |
|  | effectively as an individual, and as a | | | | |  |  |  |  |  |  |
|  | member or leader in diverse teams, and in | | | | | 3 |  | 2 |  |  |  |
|  | multidisciplinary settings. | | |  |  |  |  |  |  |  |  |
|  | | | | |  |  |  |  |  |  |  |
| **10. Communication**: Communicate | | | | | effectively |  |  |  |  |  |  |
|  | on complex engineering activities with the | | | | |  |  |  |  |  |  |
|  | engineering community and with society at | | | | |  |  |  |  |  |  |
|  | large, such as, being able to comprehend | | | | |  |  |  |  |  |  |
|  | and write effective reports and design | | | | | 3 |  | 2 |  |  |  |
|  | documentation, | | make | | effective |  |  |  |  |  |  |
|  | presentations, and give and receive clear | | | | |  |  |  |  |  |  |
|  | instructions. | |  |  |  |  |  |  |  |  |  |
|  | |  | |  |  |  |  |  |  |  |  |
| **11. Project** | | **management** | | **and** | **finance**: |  |  |  |  |  |  |
|  | Demonstrate knowledge and understanding | | | | |  |  |  |  |  |  |
|  | of the engineering and management | | | | |  |  |  |  |  |  |
|  | principles | and | apply these to | | one‘s own | 2 |  | 2 |  |  |  |
|  | work, as a member and leader in a team, to | | | | |  |  |  |  |
|  |  |  |  |  |  |  |
|  | manage projects and in multidisciplinary | | | | |  |  |  |  |  |  |
|  | environments. | |  |  |  |  |  |  |  |  |  |
|  |  | | | | |  |  |  |  |  |  |
| **12.** | **Life-long learning**: Recognize the need for, | | | | |  |  |  |  |  |  |
|  | and have the preparation and ability to | | | | |  |  |  |  |  |  |
|  | engage in independent and life-long learning | | | | | 1 |  | 3 |  |  |  |
|  | in the broadest context of technological | | | | |  |  |  |  |  |  |
|  | change. |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  | **PSOs** |  |  |  |  |  |  |  |  |  |  |
| **1.** Analyze, design and develop computing solutions by applying foundational concepts of | | | | | | | |  | 3 | 1 |  |
| computer science and engineering. | | | | |  |  |  |  |  |
|  |  |  |  |  |  |  |
| 2. Apply software engineering principles and practices for developing quality software for | | | | | | | |  | 3 | 1 |  |
| scientific and business applications. | | | | |  |  |  |  |  |
|  |  |  |  |  |  |  |
| 3. Adapt to emerging information and communication technologies (ICT) to innovate ideas | | | | | | | |  | 1 | 3 |  |
| and solutions to existing/novel problems. | | | | | |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  | 4 |  |  |  |  |  |  |

**ANNA UNIVERSITY, CHENNAI**

**AFFILIATED INSTITUTIONS**

**B.E. COMPUTER SCIENCE AND ENGINEERING**

**REGULATIONS – 2017**

**CHOICE BASED CREDIT SYSTEM**

**I & II SEMESTERS CURRICULA AND SYLLABI**

**SEMESTER I**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **SI.** |  | **COURSE** |  | **COURSE TITLE** |  | **CATEGORY** | **CONTACT** | | **L** | **T** | **P** | **C** |  |
| **No** |  | **CODE** |  |  | **PERIODS** | |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **THEORY** | | |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1. |  | HS8151 |  | Communicative English |  | HS | 4 |  | 4 | 0 | 0 | 4 |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2. |  | MA8151 |  | Engineering |  | BS | 4 |  | 4 | 0 | 0 | 4 |  |
|  |  |  |  | Mathematics - I |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3. |  | PH8151 |  | Engineering Physics |  | BS | 3 |  | 3 | 0 | 0 | 3 |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4. |  | CY8151 |  | Engineering Chemistry |  | BS | 3 |  | 3 | 0 | 0 | 3 |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5. |  | GE8151 |  | Problem Solving and |  | ES | 3 |  | 3 | 0 | 0 | 3 |  |
|  |  |  |  | Python Programming |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6. |  | GE8152 |  | Engineering Graphics |  | ES | 6 |  | 2 | 0 | 4 | 4 |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **PRACTICALS** | | |  |  |  |  |  |  |  |  |  |  |  |
| 7. |  | GE8161 |  | Problem Solving and |  | ES | 4 |  |  |  |  |  |  |
|  |  |  |  | Python Programming |  |  |  |  | 0 | 0 | 4 | 2 |  |
|  |  |  |  | Laboratory |  |  |  |  |  |  |  |  |  |
| 8. |  | BS8161 |  | Physics and Chemistry |  | BS | 4 |  | 0 | 0 | 4 | 2 |  |
|  |  |  |  | Laboratory |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | **TOTAL** | **31** |  | **19** | **0** | **12** | **25** |  |
|  |  |  |  |  | **SEMESTER II** | |  |  |  |  |  |  |  |
| **SI.No** |  | **COURSE** |  | **COURSE TITLE** |  | **CATEGORY** | **CONTACT** |  | **L** | **T** | **P** | **C** |  |
|  | **CODE** |  |  | **PERIODS** |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  | |  |  |  |  |  |  |  |  |  |  |  |  |
| **THEORY** | | |  |  |  |  |  |  |  |  |  |  |  |
| 1. |  | HS8251 |  | Technical English |  | HS | 4 |  | 4 | 0 | 0 | 4 |  |
| 2. |  | MA8251 |  | Engineering |  | BS | 4 |  | 4 | 0 | 0 | 4 |  |
|  |  |  |  | Mathematics - II |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3. |  | PH8252 |  | Physics for Information |  | BS | 3 |  | 3 | 0 | 0 | 3 |  |
|  |  |  |  | Science |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4. |  | BE8255 |  | Basic Electrical, |  | ES |  |  |  |  |  |  |  |
|  |  |  |  | Electronics and |  |  | 3 |  | 3 | 0 | 0 | 3 |  |
|  |  |  |  | Measurement |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  | Engineering |  |  |  |  |  |  |  |  |  |
| 5. |  | GE8291 |  | Environmental Science |  | HS | 3 |  | 3 | 0 | 0 | 3 |  |
|  |  |  |  | and Engineering |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6. |  | CS8251 |  | Programming in C |  | PC | 3 |  | 3 | 0 | 0 | 3 |  |
| **PRACTICALS** | | |  |  |  |  |  |  |  |  |  |  |  |
| 7. |  | GE8261 |  | Engineering Practices |  | ES | 4 |  | 0 | 0 | 4 | 2 |  |
|  |  |  |  | Laboratory |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 8. |  | CS8261 |  | C Programming |  | PC | 4 |  | 0 | 0 | 4 | 2 |  |
|  |  |  |  | Laboratory |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | **TOTAL** | **28** |  | **20** | **0** | **8** | **24** |  |
|  |  |  |  |  | 5 | |  |  |  |  |  |  |  |

**GE8151 PROBLEM SOLVING AND PYTHON PROGRAMMING**  **L T P C 3 0 0 3**

**OBJECTIVES:**

1. To know the basics of algorithmic problem solving
2. To read and write simple Python programs.
3. To develop Python programs with conditionals and loops.
4. To define Python functions and call them.
5. To use Python data structures –- lists, tuples, dictionaries.
6. To do input/output with files in Python.

**UNIT I** **ALGORITHMIC PROBLEM SOLVING** **9**

Algorithms, building blocks of algorithms (statements, state, control flow, functions), notation (pseudo code, flow chart, programming language), algorithmic problem solving, simple strategies for developing algorithms (iteration, recursion). Illustrative problems: find minimum in a list, insert a card in a list of sorted cards, guess an integer number in a range, Towers of Hanoi.

**UNIT II** **DATA, EXPRESSIONS, STATEMENTS** **9**

Python interpreter and interactive mode; values and types: int, float, boolean, string, and list; variables, expressions, statements, tuple assignment, precedence of operators, comments; modules and functions, function definition and use, flow of execution, parameters and arguments; Illustrative programs: exchange the values of two variables, circulate the values of n variables, distance between two points.

**UNIT III** **CONTROL FLOW, FUNCTIONS** **9**

Conditionals: Boolean values and operators, conditional (if), alternative (if-else), chained conditional

(if-elif-else); Iteration: state, while, for, break, continue, pass; Fruitful functions: return values, parameters, local and global scope, function composition, recursion; Strings: string slices, immutability, string functions and methods, string module; Lists as arrays. Illustrative programs: square root, gcd, exponentiation, sum an array of numbers, linear search, binary search.

**UNIT IV** **LISTS, TUPLES, DICTIONARIES** **9**

Lists: list operations, list slices, list methods, list loop, mutability, aliasing, cloning lists, list parameters;

Tuples: tuple assignment, tuple as return value; Dictionaries: operations and methods; advanced list processing - list comprehension; Illustrative programs: selection sort, insertion sort, mergesort, histogram.

**UNIT V** **FILES, MODULES, PACKAGES** **9**

Files and exception: text files, reading and writing files, format operator; command line arguments, errors and exceptions, handling exceptions, modules, packages; Illustrative programs: word count, copy file.

**OUTCOMES:**

**Upon completion of the course, students will be able to**

* Develop algorithmic solutions to simple computational problems
* Read, write, execute by hand simple Python programs.
* Structure simple Python programs for solving problems.
* Decompose a Python program into functions.
* Represent compound data using Python lists, tuples, dictionaries.
* Read and write data from/to files in Python Programs.

**TOTAL : 45 PERIODS**

**6**

**TEXT BOOKS:**

1. Allen B. Downey, ``Think Python: How to Think Like a Computer Scientist’’, 2nd edition, Updated for Python 3, Shroff/O’Reilly Publishers, 2016 [(http://greenteapress.com/wp/think-python/)](http://greenteapress.com/wp/think-python/)
2. Guido van Rossum and Fred L. Drake Jr, ―An Introduction to Python – Revised and updated for Python 3.2, Network Theory Ltd., 2011.

**REFERENCES:**

1. John V Guttag, ―Introduction to Computation and Programming Using Python’’, Revised and expanded Edition, MIT Press , 2013
2. Robert Sedgewick, Kevin Wayne, Robert Dondero, ―Introduction to Programming in Python:

An Inter-disciplinary Approach, Pearson India Education Services Pvt. Ltd., 2016.

1. Timothy A. Budd, ―Exploring Python‖, Mc-Graw Hill Education (India) Private Ltd.,, 2015.
2. Kenneth A. Lambert, ―Fundamentals of Python: First Programs‖, CENGAGE Learning, 2012.
3. Charles Dierbach, ―Introduction to Computer Science using Python: A Computational Problem-Solving Focus, Wiley India Edition, 2013.
4. Paul Gries, Jennifer Campbell and Jason Montojo, ―Practical Programming: An Introduction to Computer Science using Python 3‖, Second edition, Pragmatic Programmers, LLC, 2013.

7

|  |  |  |
| --- | --- | --- |
| **GE8161** | **PROBLEM SOLVING AND PHYTHON PROGRAMMING** | **L T P C** |
|  | **LABORATORY** | **0 0 4 2** |

**OBJECTIVES**

* To write, test, and debug simple Python programs.
* To implement Python programs with conditionals and loops.
* Use functions for structuring Python programs.
* Represent compound data using Python lists, tuples, dictionaries.
* Read and write data from/to files in Python.

**LIST OF PROGRAMS**

1. Compute the GCD of two numbers.
2. Find the square root of a number (Newton’s method)
3. Exponentiation (power of a number)
4. Find the maximum of a list of numbers
5. Linear search and Binary search
6. Selection sort, Insertion sort
7. Merge sort
8. First n prime numbers
9. Multiply matrices
10. Programs that take command line arguments (word count)
11. Find the most frequent words in a text read from a file
12. Simulate elliptical orbits in Pygame
13. Simulate bouncing ball using Pygame

**PLATFORM NEEDED**

Python 3 interpreter for Windows/Linux

**OUTCOMES**

Upon completion of the course, students will be able to

* Write, test, and debug simple Python programs.
* Implement Python programs with conditionals and loops.
* Develop Python programs step-wise by defining functions and calling them.
* Use Python lists, tuples, dictionaries for representing compound data.
* Read and write data from/to files in Python.

**TOTAL :60 PERIODS**

8