



SRI VASAVI ENGINEERING COLLEGE (AUTONOMOUS)

(Sponsored by Sri Vasavi Educational Society)

(Approved by AICTE, New Delhi & Permanently affiliated to JNTUK, Kakinada)

(Accredited by NAAC with 'A' Grade, Recognized by UGC under section 2(f) & 12(B))

Pedatadepalli, **TADEPALLIGUDEM – 534 101.W.G.Dist. (A.P)**

Department of Computer Science & Engineering (Accredited by NBA)

**Minutes of the
Fourth Board of
Studies held on
29/12/2020 at 10:30
AM through online
mode**



SRI VASAVI ENGINEERING COLLEGE (AUTONOMOUS)

(Sponsored by Sri Vasavi Educational Society)

(Approved by AICTE, New Delhi & Permanently affiliated to JNTUK, Kakinada)

(Accredited by NAAC with 'A' Grade, Recognized by UGC under section 2(f) & 12(B))

Pedatadepalli, **TADEPALLIGUDEM – 534 101**, W.G.Dist. (A.P)

Department of Computer Science & Engineering (Accredited by NBA)

Dt: 29.12.2020

The fourth meeting of Board of Studies in Department of Computer Science and Engineering is held at 10.30 AM on 29-12-2020 through online mode using,

<https://us02web.zoom.us/j/86038326979>

The following members attended the meeting:

S.No.	Name of the Member	Designation	Role
1.	Dr. D Jaya Kumari	Professor, HoD-CSE, SVEC	Chairperson
2.	Dr.Krishna Mohan Ankala	Professor, UCEK, Kakinada	University Nominee
3.	Dr. R.B.V. Subramanyam	Professor, Department of CSE, NIT Warangal	Academic Expert
4.	Sri. Srinivasa Raju Vuppalapati	Senior Consultant, MSR IT Services LLP, Hitech City, Hyderabad.	Industry Expert
5.	Mr.EEdala Rambabu	microfocus, Bangalore	Alumni
6.	Dr. V. Venkateswara Rao	Professor	Member
7.	Dr. G Loshma	Associate Professor	Member
8.	Ch. Raja Ramesh	Associate Professor	Member
9.	Dr. V S Naresh	Associate Professor	Member
10.	Dr.K. ShirinBhanu	Associate Professor	Member
11.	A. Leelavathi	Assistant Professor	Member
12.	R. LeelaPhani Kumar	Assistant Professor	Member
13.	G. Nataraj	Assistant Professor	Member
14.	B.SriRamy	Assistant Professor	Member
15.	G.Sriram Ganesh	Assistant Professor	Member
16.	N.V.Murali Krishna Raja	Assistant Professor	Member
17.	N. Hiranmayee	Assistant Professor	Member
18.	Y.DivyaVani	Assistant Professor	Member
19.	M NageswaraRao	Assistant Professor	Member
20.	B Kiran Kumar	Assistant Professor	Member
21.	Y. Ravi Raju	Assistant Professor	Member
22.	D.S L Manikanteswari	Assistant Professor	Member
23.	M. Anantha Lakshmi	Assistant Professor	Member
24.	M. Satyanarayana Reddy	Assistant Professor	Member
25.	J.N. Chandra Sekhar	Assistant Professor	Member
26.	P. Bhavani Shankar	Assistant Professor	Member
27.	David Raju. K	Assistant Professor	Member
28.	P Rajesh	Assistant Professor	Member
29.	P Suneetha	Assistant Professor	Member
30.	P Laxmi kanth	Assistant Professor	Member
31.	K Satyanarayana	Assistant Professor	Member
32.	M Sree Radha Mangamani	Assistant Professor	Member
33.	S K Shabuddin	Assistant Professor	Member
34.	G Ramanjaneya Raju	Assistant Professor	Member
35.	G V Lakshmi Narayana	Assistant Professor	Member
36.	A Nageswara Rao	Assistant Professor	Member
37.	Mr. L Balaji	Lecturer	Member

The following are the Minutes of the Meeting

Item No.1: Welcome note by the Chairperson BOS.

The HOD extended a formal welcome and introduced the members.

Item No.2: Review & Approval of the Course Structure for I and II SEM -B.Tech (CSE) and B.Tech (CST) Programme under V20 Regulation.

Reviewed the Course Structure of I & II semesters for B.Tech (CSE) and B.Tech (CST) Programme of V20 Regulation. The approved Syllabus is given in **Annexure-I**.

Item No.3: Approval of Syllabi for the proposed courses offered in I and II SEM -B.Tech (CSE) and B.Tech (CST) Programme under V20 Regulation.

- Approved the syllabi for the courses offered in I & II semesters B.Tech (CSE) and B.Tech (CST) Programme of V20 Regulation. The approved Syllabus is given in **Annexure-II**.

-

Dr.D Jaya Kumari
Chairperson of BOS

Head of the Department
Dept. of Computer Science & Engineering
Sri Vasavi Engineering College
TADEPALLIGUDEM-534 101

Annexure-I

I SEMESTER (First Year)

S.No.	Category	Code	Course Title	L	T	P	C
1	BSC	V20MAT01	Linear Algebra and Differential Equations	3	-	-	3
2	BSC	V20CHT01	Engineering Chemistry	3	-	-	3
3	HSS	V20ENT01	English for Professional Enhancement	3	-	-	3
4	ESC	V20MEL02	Engineering Workshop	1	-	4	3
5	ESC	V20CST01	Programming in 'C' for problem Solving	3	-	-	3
6	HSS LAB	V20ENL01	Hone Your Communications Skills Lab-I	-	-	3	1.5
7	BSC LAB	V20CHL01	Engineering Chemistry Lab	-	-	3	1.5
8	ESC LAB	V20CSL01	Programming Lab in 'C' for problem Solving	-	-	3	1.5
Total credits							19.5

Category	Credits
Basic Science course	7.5
Engineering Science Courses	7.5
Humanities and social science	4.5
TOTAL CREDITS	19.5

II SEMESTER (First Year)

S.No.	Category	Code	Course Title	L	T	P	C
1	BSC	V20MAT02	Numerical Methods and Vector Calculus	3	-	-	3
2	BSC	V20PHT01	Engineering Physics	3	-	-	3
3	ESC	V20ECT01	Switching Theory and Logic Design	3	-	-	3
4	ESC	V20CST02	Python Programming	3	-	-	3
5	ESC	V20MEL01	Engineering Graphics	1	-	4	3
6	ESC LAB	V20PHL01	Engineering Physics Lab	-	-	3	1.5
7	BSC LAB	V20CSL02	Python Programming Lab	-	-	3	1.5
8	HSS LAB	V20ENL02	Hone Your Communications Skills Lab-II	-	-	3	1.5
9	MNC (AICTE suggested)	V20CHT02	Environmental Studies	2	-	-	0
Total credits							19.5

Category	Credits
Basic Science course	7.5
Engineering Science Courses	10.5
Humanities and social science	1.5
TOTAL CREDITS	19.5

Annexure-II

Semester	I /II SEM	L	T	P	C	COURSE CODE
Regulation	V20	-	-	3	1.5	V20CST01
Name of the Course	Programming in 'C' for problem Solving					
Branches	Common to All					

Course Outcomes:

After successful completion of this course, the students will be able to

CO No.	Course Outcome	Knowledge Level
CO1	Describe various problem solving strategies such as Algorithms and Flowcharts	K2
CO2	Develop various programming constructs using Control Structures	K3
CO3	Construct Programs using modular programming approach	K3
CO4	Illustrate the usage of Arrays, String and pointers	K3
CO5	Construct Programs using Structures and Unions	K3
CO6	Distinguish between Sequential files and Random access files	K4

UNIT-I: Problem solving concepts: Algorithms, Flow-charts, Types of Programming Languages, Compiler, Assembler and Linker, Testing and Debugging a program. **Introduction to C Programming:** Overview and importance of C, C Program Structure, Creation and Compilation of C Programs, Identifiers, Variables, Data types, Constants, Declarations, **Input and output statements:** Input and output functions..

UNIT-II: Operators: Arithmetic, relational and logical operators, increment and decrement operators, conditional operator, assignment operator, bitwise operators, special operators, expressions, Precedence, Associativity, Order of evaluation, Type conversion, Programming Examples. **Control Structures:** Conditional statements - If-else, Switch-case constructs, Loops - while, do-while, for.

UNIT-III: Functions: Top down approach of problem solving, standard library functions, user defined functions, parameter passing - call by value, call by reference, return statement, passing arrays as parameters to functions, recursion. **Storage Classes:** Scope and extent, Storage Classes - auto, extern, static and register.

Understanding pointers: Accessing the address of a variable, declaring pointer variables, initialization of pointer variables, accessing a variable through its pointer, pointer arithmetic.

UNIT-IV: Arrays: Single-Dimensional Arrays, multi-Dimensional Arrays, initialization and accessing individual elements. **Strings** in C- Concepts, string handling functions. Pointer and arrays, pointers and character strings, array of pointers. **Dynamic Memory Allocation:** calloc(), malloc() and free()

UNIT-V: Structures: Defining, declaring, initialization, accessing, comparing, operations on individual members, array of structures, structures within structures, structures and functions, bit fields, Programming Examples. **Unions:** Definition – difference between structures and unions – declaring and accessing unions.– pointers and structures – self-referential structures.

UNIT-VI: File Processing: Creating and Opening a file, file opening modes, closing a file, input/output operations on files, error handling during I/O operations, random access to files, Command line arguments. Programming Examples.

TEXT BOOKS:

1. Programming in ANSI C by E Balagursamy, McGraw Hill, 8th Edition.

REFERENCE BOOKS:

1. Let Us C, [YashavantKanetkar](#), BPB Publications, 15th Edition
2. Programming in C, ReemaThareja, Oxford.
3. Programming with C, Second edition, Byron S Gottfried, Tata McGrawhill
4. Problem Solving and Programming design in C, Hanly J R &Koffman E.B, Pearson Education, 2009.
5. Programming in C, PradipDey, ManasGhosh, Oxford University Press, 2007.
6. Problem Solving Using C: Structured Programming Techniques, [YukselUckan](#).
7. C Programming, A Problem Solving Approach, Forouzan, Gilberg, Prasad, CENGAGE.
8. Computer Programming in C – Kerningham& Ritchie, PHI
9. C: The Complete Reference: Herbert Schildt, Osborne/Mcgraw Hill, Inc.

Semester	I/II SEM	L	T	P	C	COURSE CODE
Regulation	V20	-	-	3	1.5	V20CSL01
Name of the Course	Programming Lab in 'C' for problem Solving					
Branches	Common to All					

Course Outcomes:

After successful completion of this course, the students will be able to

CO No.	Course Outcome	Knowledge Level
CO1	Demonstrate problem solving techniques using Control Structures	K3
CO2	Construct Programmes using the concepts of Arrays, Strings and Pointers	K3
CO3	Apply the concepts of Functions, Structures and Unions	K3
CO4	Use various file processing operations to develop real-time applications	K4

LIST OF EXPERIMENTS

Tutorial 1: Problem solving using computers.

Lab1: Familiarization with programming environment.

Tutorial 2: Variable types and type conversions.

Lab 2: Simple computational problems using arithmetic expressions.

Tutorial 3: Branching and logical expressions.

Lab 3: Problems involving if-then-else structures switch – case.

Tutorial 4: Loops, while and for loops.

Lab 4: Iterative problems e.g. sum of series.

Tutorial 5: Functions call by value, call by reference

Lab 5: Simple functions.

Tutorial 6: Recursion, structure of recursive calls.

Lab 6: Recursive functions.

Tutorial 7: Pointers.

Lab 7: Programming with pointers.

Tutorial 8: 1D Arrays: searching, sorting.

Lab 8: 1D Array manipulation.

Tutorial 9: 2D arrays.

Lab 9: Matrix problems.

Tutorial 10: String handling.

Lab 10: String handling functions.

Tutorial 11: Structures, unions and dynamic memory allocation.

Lab 11: Structures & unions.

Tutorial 12: File handling, command line arguments.

Lab 12: File operations.

TEXT BOOKS:

1. Programming in Ansi C by E Balagursamy, McGraw Hill, Eight Edition.

Reference Books:

1. Mastering C, K.R. Venugopal and S.R. Prasad, TMH Publishers.
2. Computer Programming in C, V. Rajaraman, PHI.
3. Programming in C, Stephen G. Kochan, Fourth Edition, Pearson Education.
4. C- The Complete Reference, Herbert Schildt, Osborne/Mcgraw Hill, Inc.
5. Programming with C, Byron S Gottfried, Second edition, Tata McGrawhill.
6. Programming in C, ReemaThareja, Oxford.
7. Problem Solving and Program design in C, Hanly J R &Koffman E.B, Pearson Education, 2009
8. Programming and Problem Solving Using C, ISRD Group, Tata McGraw Hill,2008

Semester	II SEM	L	T	P	C	COURSE CODE
Regulation	V20	3	-	-	3	V20CST02
Name of the Course	PYTHON PROGRAMMING					
Branches	Common to CSE & CST					

Course Outcomes:

After successful completion of this course, the students will be able to

CO No.	Course Outcome	Knowledge Level
CO1	Illustrate basic concepts of Python Programming	K2
CO2	Describe control structures in python	K2
CO3	Construct python programs using structured data types	K3
CO4	Demonstrate functions and packages	K3
CO5	Compare TextFiles and Binary Files	K4
CO6	Apply OOPs concepts to Develop Test cases	K3

Syllabus

UNIT-I: Introduction to Python, Data Types & Operators: Basics of python programming: Features of python – History of Python - Python installation and execution - Data types – Identifiers - variables – type conversions- Literals, Constants – Numbers – Strings. I/O statements.Operators and expressions, operator precedence – expression evaluation.

UNIT-II: Control Structures: Decision Control statements: conditional (if), alternative (if-else), chained conditional (if-elif-else); **Iteration:** while loop, for loop, nested for loop, range function, break, continue and pass statements.

UNIT-III: Structured Data Types: Lists: list operations, list slices, list methods, cloning lists, list parameters. **Tuples:** tuple assignment, tuple as return value. **Set:** Set Creation, Set Operations. **Dictionaries:** Creation, operations; comprehension, operations on strings.

UNIT-IV: Functions& modules: Introduction - Function Declaration &Definition - Function Call – Variable Scope and Lifetime - The return statement-More on Defining Functions - Lambda Functions or Anonymous Functions - Documentation Strings- Modules – Packages.

UNIT-V: Files & Exception Handling: Introduction - Types of files - Text files - reading and writing files; Errors and exceptions handling.

UNIT-VI:OOPS concepts Classes, Methods, Constructor, Inheritance, Overriding Methods, Data hiding, TKINTER.

Text Books:

1. “Python Programming using problem solving Approach” ReemaThareja, Oxford University Press – 2017.
2. Python with Machine Learning by “A.Krishna Mohan, Karunakar&T.Murali Mohan” by S. Chand Publisher-2018.

Reference 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>)
2. Guido van Rossum and Fred L. Drake Jr, –An Introduction to Python – Revised and updated for Python 3.2, Network Theory Ltd., 2011.
3. John V Guttag, –Introduction to Computation and Programming Using Python“, Revised and expanded Edition, MIT Press , 2013

Semester	II SEM	L	T	P	C	COURSE CODE
Regulation	V20	-	-	3	1.5	V20CSL02
Name of the Course	PYTHON PROGRAMMING LAB					
Branches	Common to CSE & CST					

Course Outcomes:

After successful completion of this course, the students will be able to

CO No.	Course Outcome	Knowledge Level
CO1	Demonstrate Basic Python Programs	K3
CO2	Construct control structures in python	K3
CO3	Demonstrate functions and packages	K3
CO4	Construct python programs using structured data types	K3
CO5	Demonstrate Text Files and exception handling	K3
CO6	Test Rock – paper – Scissors game	K4

Syllabus

Exercise 1 - Basics

- A sample Python Script using command prompt, Python Command Line and IDLE
- A program to purposefully raise an Indentation Error and correct it

Exercise 2 - Operations

- A program to compute distance between two points taking input from the user (Pythagorean Theorem)
- A program on add.py that takes 2 numbers as command line arguments and prints its sum.

Exercise - 3 Control Flow

- A Program to implement for checking whether the given number is a even number or not.
- A program to construct reverse the digits of a given number and add it to the original, If the sum is not a palindrome repeat this procedure.
- A program using a while loop that asks the user for a number, and prints a countdown from that number to zero.

Exercise 4 - Control Flow – Continued

- A program to construct the following pattern, using a nested for loop.

```

*
* *
* * *
* * * *
* * * * *
* * * *
* * *
* *
*

```
- By considering the terms in the Fibonacci sequence whose values do not exceed four million, find the sum of the even-valued terms.

Exercise - 5 Structured Data types

- a) A program to count the number of strings where the string length is 2 or more and the first and last character are same from a given list of strings.
- b) a program to develop unzip a list of tuples into individual lists and convert them into dictionary.

Exercise - 6 Structured Data types Continued

- a) A program to count the numbers of characters in the string and store them in a dictionary data structure
- b) A program to use split and join methods in the string and trace a birthday with a dictionary data structure.

Exercise - 7 – Problem Solving using Functions

- a) Find mean, median, mode for the given set of numbers passed as arguments to a function
- b) Develop a function nearly_equal to test whether two strings are nearly equal. Two strings a and b are nearly equal when a can be generated by a single mutation on b.
- c) Develop a Recursive Function to find the Factorial of a given number.
- d) Develop function to compute gcd, lcm of two numbers. Each function shouldn't exceed one line.

Exercise - 8– Modules

- a) Install packages requests, flask and explore them using (pip)
- b) A program to implement a script that imports requests and fetch content from the page. Eg. (Wiki)
- c) Develop a simple script that serves a simple HTTPResponse and a simple HTML Page

Exercise - 9 Files

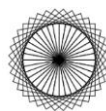
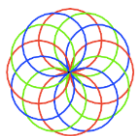
- a) A program to count frequency of characters in a given file. Can you use character frequency to tell whether the given file is a Python program file, C program file or a text file?
- b) A program to compute the number of characters, words and lines in a file.

Exercise - 10 OOP

- a) Class variables and instance variable and illustration of self variable
 - i) Robot
 - ii) ATM Machine

Exercise - 11 GUI, Graphics

1. Develop a GUI for an Expression
2. A program to implement the following figures using turtle



Text Books:

1. “Python Programming using problem solving Approach” Reema Thareja, Oxford University Press – 2017.
2. Python with Machine Learning by “A.Krishna Mohan, Karunakar & T.Murali Mohan” by S. Chand Publisher-2018.