F 3187   (Pages : 2)   Reg. No................................

Name.................................

B.TECH. DEGREE EXAMINATION, NOVEMBER 2014

Fifth Semester

Branch : Civil Engineering

COMPUTER PROGRAMMING (C)

(Old Scheme—Supplementary/Mercy Chance)

[Prior to 2010 Admissions]

Time : Three Hours   Maximum : 100 Marks

Write neat and efficient C programs wherever necessary.

Part A

Answer all questions briefly.
Each question carries 4 marks.

1. What is linker ? Why it is required ?

2. List the characteristic features of high level languages. Name any four high level languages.

3. Distinguish between variables, constants and keywords with suitable examples.

4. Describe the basic data types and specify their size and range.

5. What are functions ? Write the general syntax for declaring a function.

6. What is the necessity of a "return" statement ? When is it necessary and when is it a must ?

7. Under what conditions can one pointer variable be compared ? Under what conditions are such comparisons useful ?

8. What happens when a pointer to a structure is incremented ?

9. What are the three steps that are followed while accessing a file ?

10. Write the statements needed for file operations in C, and show how a sequential file can be opened and read.

(10 × 4 = 40 marks)

Part B

Answer all questions.
Each full question carries 12 marks.

11. (a) What is a file extension in a file name ? What for are file extensions typically used ? List some typical file extensions and their meanings.

(6 marks)

(b) Describe the various steps in programming using C language.

(6 marks)

Or

Turn over

12. Write the algorithm and draw neat flow chart to test whether a given number is prime or not.

13. (a) For the printf() function, what are the format specifiers for specifying the types of argument, the position and the precision of the output ? Give suitable examples.

(8 marks)

(b) What are the differences between scanf() and gets() with respect to data input with embedded white space characters ?

(4 marks)

Or

14. Draw the flow chart and write the C program to generate and print the first N Fibonacci numbers.

15. Write a function which checks whether a matrix is orthogonal or not.

Or

16. Explain with program examples, the different methods of passing arguments to a function.

17. Write a C program to create a linear linked list of nodes containing roll numbers and interactively print out the list and total number of items in the list.

Or

18. Read in the first *n* elements of a one-dimensional floating point array. Calculate the sum of these elements, the mean and standard deviation.

19. Write a C program to reverse the contents of a text file.

Or

20. A student master file consists of the register number, name and marks in the 8 subjects. Write a C program which will read the file and print a list of students who have failed in one or more subjects. The pass minimum required is 40 % in each subject.

(5 × 12 = 60 marks)