

# Programming & Data Structure Laboratory (CS112)

End-Semester Exam  
Indian Institute of Technology, Patna  
June 30, 2022

Submission Deadline: 4:30 pm 30/06/2022

## Instructions:

- Proper indentation is mandatory.
- You should comment the statements whenever necessary, to make the code understandable.
- Answers files: **endsem\_1.c**, **endsem\_2.c**

**Total Marks: 50**

**DO NOT COPY: BOTH SOURCE AND DESTINATION WILL GET ZERO**

**EXERCISE 1:** A decimal number  $n$  between 0 and 32 exclusive ( $0 \leq n < 32$ ) can be expressed in binary system as  $x_4x_3x_2x_1x_0$ , where  $x_i$ 's are either zero or one. Write a C program that accepts (from the terminal) a decimal number in the above range and prints out the equivalent binary representation. [20]

### Test data and expected output:

Enter a +ve number less than 32: -5  
Entered number is out of range

Enter a +ve number less than 32: 21  
Binary equivalent of decimal number 21 is 10101

Enter a +ve number less than 32: 14  
Binary equivalent of decimal number 14 is 01110

Enter a +ve number less than 32: 35  
Entered number is out of range

**EXERCISE 2:** Let there be a set of disks A,B,C,D,E . Each one is pushed and popped in a stack in following order PUSH(A), PUSH(B), POP(), PUSH(C), PUSH(D), POP(),POP(), POP(), PUSH(E), POP(). Each operation push and pop takes 1 unit of time. Calculate time of each disk to stay in the stack. Your program should work with any sequence of PUSH and POP and any number of disks. Assume that any disk will be pushed and popped at most only once. First, your program will take the sequence of push and pop from the user. For the above example the sequence input is ABPCDPPPEP, where "P" stands for pop operation and any other character stands for push operation of a disk denoted by that character. Assume that there is no disk as "P". Your program should print an appropriate error message in case of any invalid input sequence. [30]

### Example Output:

Input sequence of operations: ABPCDPPPEP  
Push time of A: 1  
Push time of B: 2  
Pop time of B: 3  
Total Time of B in stack: 1  
Push time of C: 4  
Push time of D: 5  
Pop time of D: 6  
Total Time of D in stack: 1  
Pop time of C: 7  
Total Time of C in stack: 3

Pop time of A: 8  
Total Time of A in stack: 7  
Push time of E: 9  
Pop time of E: 10  
Total Time of E in stack: 1