CS559: Computer Systems Lab

READ THE FOLLOWING INSTRUCTIONS CAREFULLY. FAILING TO ADHERE TO IT WILL LEAD TO DEDUCTION OF MARKS.

1. Make sure to give the correct information in your submission link. You will receive a response email for every submission. **You must save it for future reference for the whole semester as you will be asked to show it.**

2. Every assignment will have multiple questions. **For every question, you will save your solution (i.e. program) as a .c file.** So if "Assignment 01" has 4 questions (marked as 1, 2, 3 and 4), then you will save your solution program as 1.c, 2.c, 3.c and 4.c.

   Note: Your program will be compiled to check if your solution is correct and you will be marked accordingly. If any solution file is missing or not submitted, the marks for that solution will be zero.

3. Submit your files in a compressed folder (rar or zip) as per "submission instructions" given in your assignment question paper.

   Note: You must make sure to upload in .zip or .rar. **Folders submitted with any other extension or wrong extension will not be evaluated and awarded** zero.

4. The student must ensure that the submitted file is not corrupted and can be unzipped properly. **Corrupted files that cannot be opened will be given** zero. **Similarly bad filenames having any other extension than ".c" will not be evaluated and given** zero.

5. You will be able to upload your assignment multiple times within the deadline span, but only your last submission will be treated as final and considered for evaluation. **No plea/request to consider intermediate submissions for evaluation will be entertained.** So, make sure to carefully check that you are submitting the correct assignment.

6. The submission link will be automatically deactivated after the deadline, and no request/plea for extension will be accepted. **Non-submission/wrong submission will be automatically awarded** zero.

7. Your code will be checked for similarity and you will be penalized according to the following rule:

   Similarity above 75%: 50% deduction
   Similarity of 100%: full deduction

8. **Compiler Information: Program must be compiled using online gdb compiler. If your program can not be compiled using gdb compiler then marks will be reduced.**
   Compiler Link : `https://www.onlinegdb.com/online_c_compiler`

9. Marks for every assignment (other than practice assignments) will be mailed to you with remarks. **You will be allowed a deadline for rebuttal, after which no rebuttal will be entertained.**

Submission Instruction: Store your assignments in folder and compress it as a rar/zip file (filename should be in this format: roll-number_assign5.rar or roll-number_assign5.zip). For example, if your roll number is 2211CS01, store your assignment as 2211CS01_assign5.rar or 2211CS01_assign5.zip. Also, save each program in the format given beside each question. Upload the same at the below link:
`https://forms.gle/Fg8QrsXzntJJvm8P8`

Problem 1 (Save as 1.c) :- Jack has a sack with N different items in it. Each item has a cost C. Now, Jack wants to withdraw some items from the sack such that the sum of the cost of the items is equal to a fixed integer value T. Jack can also use the same item multiple times. [4 points]

Your work in this problem to is to print the list of all the unique possible combinations of items with the total cost equal to T using the recursion function (Two combinations are unique if the frequency of at least one of the chosen numbers is different.). Print error if there is no possible combination.

(Note: You may use any order of items to print values. But {2,3,4} and {3,2,4} both will be considered same.)

Example: Suppose, Sack has 4 items with each having cost of 20\$,30\$,40\$, and 80\$. Jack wants to withdraw items with a total cost of 80\$.

Input:
Number of items in the sack: 4
Cost of each item: 20 30 40 80
Cost of withdrawal: 80

Expected Output:
Number of unique lists of items with a total cost 80: 5
List of all the unique possible combinations of items: [20,20,20,20], [20,30,30], [20,20,40], [40,40], [80].

Problem 2 (Save as 2.c) :- Write a C program to swap two numbers using a third variable using **assembly level instructions only**. (**Note:** Inline assembly using __asm__ keyword is to be used. Check supplementary material.) [3 points]

Input:
Declare two integer variables **arg1** and **arg2** and assign them **51** and **15** respectively (Example: int arg1=51, arg2=15;).

Expected Output:
arg1 = 15
arg2 = 51