

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.**

CS559: Computer Systems Lab

Date: [10 th September,2022]

Assignment [5]

Submission deadline: [17th September, 2023 - 1800 Hrs. IST]

Submission Instruction: Store your assignments in folder and compress it as a rar/zip file (filename should be in this format: `roll-number_assign(Assignment Number).rar` or `roll-number_assign[Assignment Number].zip`).

For example, if your roll number is 2211CS01, store your assignment as 2211CS01_assign[5].rar or 2211CS01_assign[5].zip.

Also, save each program in the format given beside each question. Upload the same at the below link:

<https://forms.gle/CPmEx2eGmAcMJS3E6>

Problem [1] (Save as per extension) :- You are working with a simplified computer architecture that employs zero-address instructions. In this architecture, all instructions are single-word commands that operate on a stack. The stack contains integer values, and the instructions are executed sequentially. [10 points]

Here are the available instructions:

- **PUSH n:** Push the integer n onto the stack.
- **POP:** Pop the top value from the stack and discard it.
- **ADD:** Pop the top two values from the stack, add them together, and push the result back onto the stack.
- **SUB:** Pop the top two values from the stack, subtract the second from the first, and push the result back onto the stack.

Write a program that takes a sequence of zero-address instructions as input and simulates their execution. Your program should start with an empty stack and execute the instructions in the given order.

1 Structure of Input

- The first line contains an integer N ($1 \leq N \leq 100$) representing the number of instructions.
- The next N lines contain the instructions. Each instruction is one of the following formats:
 - **PUSH n** where n is an integer ($-1000 \leq n \leq 1000$).
 - **POP**
 - **ADD**
 - **SUB**

2 Output

For each POP instruction, output the value that was popped from the stack. After processing all instructions, print the final content of the stack, from the top to the bottom, one value per line.

3 Sample Input Example

3.1 Input

- 6
- PUSH 10
- PUSH 5
- ADD
- PUSH 3
- SUB
- POP

3.2 Sample Output

- 12

CS559: Computer Systems Lab

Date: [10 th September,2022]

Assignment [5]

Submission deadline: [17th September, 2023 - 1800 Hrs. IST]

Problem [1]:

- Improper indentation: -1 points
- Compilation error: -1 points
- Input not taken(i.e hard-coded): -2 points
- Wrong logic/output: -6 points
 - Wrong Output: -6