

Lab Test - Instructions

1. Lab test 1: **Week 8**, 14 September.
2. This is a closed book test.
3. The question paper contains 7 sections, A, B, C, D, E, F, G.
4. You are required to answer a total of 10 questions. The questions to be answered in the lab test will be randomly chosen during the lab test session. However, the difficulty of each question paper will be made consistent among all question papers.
5. Each question carries 10 marks. Total marks = 100. Total time = 2 hrs.
6. Program templates for the questions will be given. You should use the program templates for your program development and fill in the missing code in the template.
7. If you need a piece of rough paper to draft your code, please ask the lab technician for it.
8. **Put your electronic device (such as hand phone) inside your bag, and leave them on the floor. Do not use any of your mobile electronic devices during the lab session.**
9. **Please note that the lab test session will be monitored by CCTV. We caught 2 cheating cases last semester.**

Lab Test – Note on Coding

- Note that no C++ code will be accepted. For example, no inline variable declaration (e.g. for `(int i=0; i<10; i++) ...`) is allowed as in C++ or Java. All variable declaration should be made at the beginning of the function. The compiler used for compiling your program will be a **standard C compiler**. You will get 0 mark if the compiler is unable to compile your code.
- **Do not** use **`fflush()`**. Instead you should use the **`scanf("%c",&dummychar);`** to get rid of any remaining character (such as `'\n'`) in the buffer.
- Reason – we are using an online C compiler which is unable to take in non-standard C code such as `fflush()`, etc.
- Also note that for questions asking for recursive function, if you give an iterative version, you will get 0 mark. The online mark checking component will be able to identify whether your function is recursive or not.

Lab Test – During the Test

- Use Code::Block to edit, compile and run your code for the questions. The **template file** is also provided in the system. Do not change the contents in the main() function.
 - After the program has successfully compiled and run, you may then enter your code into the system. Copy and Paste your source code into the missing code section of the question template interface. Again, please do not change the main() function code in the program template.
 - After entering the code, you may (1) Test with sample input; (2) Try compilation; (3) Run Input.
 - Save the code of a question whenever you have successfully solved the question.
 - When finished coding on all the questions, you may then “**End Test**”.
- Please make sure that you submit your code successfully to the APSA System before you leave.

Lab Test: Marking

- Marking will be done automatically using test cases. Marks will be awarded according to the correctness of each test case.
- Your submitted source code will be marked by our automatic C code marking component of the Automated Programming Submission and Assessment (APSA) System according to the correctness of your program.
 - If the question code is **unable** to **compile**, you will get 0 mark for that question. Writing programs that are able to be compiled is important.
- After all the marking, you will then be notified. You can then log into the APAS System to view the marking results.
- If you have any concerns about the correctness of the marking, please send in your request (via email) to our lab staff Ms Eng Hui Fang (ASHFEng@ntu.edu.sg) , she will collate all the requests and pass them to the lecturer Dr HUI Siu Cheung for code checking again.