Introduction to Programming (Adv)

School of Computer Science, University of Sydney

Copyright Warning

Review

Examination format and tips

Python bits

base data types and their operators (int, float, str)

list and its operators/behaviour

classes, instance variables, methods

exceptions

memory model (diagrams)

iterators

python internals, modules

C bits

base data types and their operators (int, float, char)

pointer type (scalar and non-scalar)

pointer arithmetic, sizeof

struct, union, array

address (&), indirection (*) operators

memory model (diagrams)

Language Agnostic

Control flow

floating point numbers as an approximation

Array

Idioms

Function memory (stack)

Concept of a file

Testing

Iterator

Recursion

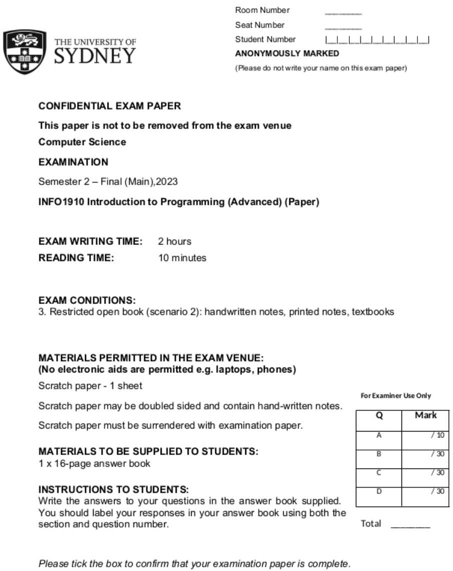
Examination format

Formal exam period

```
INFO1910
Introduction to Programming (Advanced)
Paper-based Exam
Wednesday 22/11/2023
5:00pm
120 + 10
Carslaw SR 350. Building F07.
```

Double and triple check the time and venue on the day!

Examination format



THE UNIVERSITY OF SYDNEY

Room Number ________

Seat Number ________

Student Number |__|__|__|__|__|__|__|__|__|

ANONYMOUSLY MARKED

(Please do not write your name on this exam paper)

CONFIDENTIAL EXAM PAPER

This paper is not to be removed from the exam venue

Computer Science

EXAMINATION

Semester 2 – Final (Main),2023

INFO1910 Introduction to Programming (Advanced) (Paper)

EXAM WRITING TIME: 2 hours

READING TIME: 10 minutes

EXAM CONDITIONS:
3. Restricted open book (scenario 2): handwritten notes, printed notes, textbooks

MATERIALS PERMITTED IN THE EXAM VENUE:
(No electronic aids are permitted e.g. laptops, phones)

Scratch paper - 1 sheet

Scratch paper may be doubled sided and contain hand-written notes.

Scratch paper must be surrendered with examination paper.

MATERIALS TO BE SUPPLIED TO STUDENTS:
1 x 16-page answer book

INSTRUCTIONS TO STUDENTS:
Write the answers to your questions in the answer book supplied. You should label your responses in your answer book using both the section and question number.

For Examiner Use Only

Q	Mark
A	/ 10
B	/ 30
C	/ 30
D	/ 30

Total ________

Please tick the box to confirm that your examination paper is complete.

Examination format

Read the front page before attending. Know your SID.

Section A: 10 Short answer

Section B: 30 Short programs

Section C: 30 C programs (Pick any 2 of 3 to answer)
(15, 15, 15)

Section D: 30 Python programs (Pick any 2 of 3 to answer)
(15, 15, 15)

Scratch paper

Double sided A4, very limited space

Very few references in the exam, mostly hinted for the question. Otherwise, bring your own

You have to give it up (take a photo of it before exam!)

References

Normally an examination is done without extensive access to reference material. It is important to note that answering a question may require explanation, or reasoning about why the solution may not work.

Deep understanding of function behaviour is not expected for very obscure cases.
e.g. does fgets include or exclude the new line character?

Whether or not you know, an answer can be produced on the assumption that there is a new line or there is not a new line.

Please include a comment stating that you are unsure how function deals with such cases and make an assumption. This is helpful if your program fails to function in a certain way and cannot research this function within the time available (the man pages can be very long!)

References (cont.)

Warnings

If you make wild assumptions about concepts/behaviour, you will lose marks.
e.g. I assume fgets always reads the entire file into my buffer → very wrong!.
e.g. there is always enough memory → very wrong!.

There are also no magic functions that you can assume exist to solve the problem. The scope of which functions to use is restrictive as illustrated in the question. e.g.

```
import solution
solution.run_for_full_marks()
```

Restrictions

Examples of restrictions.

C

- You may not use dynamic memory, variable length arrays, or an intermediate file to store data.
- Your program must use a constant amount of memory. You may not allocate an array with size greater than 128 bytes.

Python

- You may only use the sys library, you may not import any other Python libraries.
- You may not use the try or except keywords.

Exam is testing your ability to implement such operations either as to replicate, or to consider the design of such a solution within the context of the problem.

Tips

Problem solving:

- conceive a combination of control flow,
- implement and/or select most appropriate idiom(s)
- solve new problems using techniques of those already solved in this course
- identifying error cases, handling of errors, having a good understanding of memory models

Mechanical parts

- after designing a solution, consider functions needed
- after designing a solution, consider parameters needed
- after designing a solution, consider which data types and their operators
- after designing a solution, produce the correct output in the normal cases and the error cases (control flow).

Final grade

To pass this course, you need to:

- Obtain at least 40% in all other assessments that are not the final examination, AND
- Obtain at least 40% in the final examination, AND
- Obtain at least 50% final mark overall [1]

[1] any questions?

Teaching staff

⭐ Richard McKenzie ⭐

⭐ Simon Dowd ⭐

Special mentions

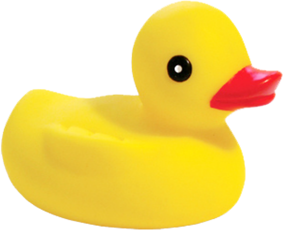
Alan Robertson
Christopher Irving
Tyson Thomas
Simon Florian Koch
Andrew Xu
Cindy Leong
Michelle Wang

Are you a real programmer yet?

If you were happy with your tutor, say it in the survey, they would like to know too!

https://student-surveys.sydney.edu.au/students/

Thank you



Good luck