



THE UNIVERSITY OF
**WESTERN
AUSTRALIA**

Last Lecture

Exam overview and discussion

Objectives

- Reminder to complete surveys
- CITS1401 Final Exam
- Questions and answers about the material covered in CITS1401
- Course summary
- CITS1401 exam 2017 (in the last lecture/workshop)

Student survey

- SELT: Student Experience of Learning and Teaching
- Email will be sent to all
- Please complete the survey
- Your feedback is important for us
 - *whether you found it useful and fair*
 - *if there is anything we could have done better*

Course Outcomes (First lecture slide)

- Developing computational thinking skills:
 - *Decompose*: how to divide large problem into small parts and solve them “*divide and conquer*”.
 - *Pattern recognition*: recognizing common tactics to solve set of problems.
 - *Abstraction*: generalizing the problem by reducing avoidable details.
 - *Algorithm*: how to formulate ordered step-wise approach to solve the problem.
- Developing programming skills:
 - *Be able to write a program in Python 3 to:*
 - Solve small problems
 - Automate repetitive computational tasks

Both skills are transferrable

Learning Outcomes

- Create algorithms using **computational thinking** to solve a range of problems
- Write **programs** using a high-level programming language Python 3 to implement algorithms including programs for *data retrieval, extraction, conversion, aggregation, calculation, processing and storage*
- Demonstrate the process of **computational problem solving** and be familiar with common approaches for computational problem solving

EXAM Sem-1 2024

- Total 7 questions
 - *not all the same size*
 - *Range of difficulty*
- Total: 100 marks
 - *Try to estimate a mark a minute, 20 minutes for revision*
- Weight in your total marks is **55%**
- Recall that
 - *Project 1* = 15%
 - *Project 2* = 20%
 - *Labs* = 10%

Exam Pattern

- Programming questions similar to previous years with differences in the questions.
- All questions should be answered.
- Write your answers in the spaces provided in the exam booklet. Each question has its own space.
- It is a closed book or notes exam.
- Exam will be Face-2-Face only and all arrangements are made by Exam Office. Contact the office if you have any queries.
- Contact with unit coordinator is not allowed during the exam.
- **No modules are allowed to be imported for any question.**

Material for exam

- Lectures
- Book (only those chapters that were covered in the lectures except math and graphics libraries)
- All laboratory sheets
- Both projects

How to prepare for the exam

1. Have a Thonny session open when you are studying for the exam
2. Go through the lecture slides
3. Consult the textbook (or links on Moodle)
 - *Try questions there too (or any other source of questions)*
4. Watch the recorded lectures again if required
5. Go through the labs again. You can only access your attempts
6. Attempt a sample exam and mark it yourself using the provided answers
7. Repeat step 1 to 6

Unit summary

- Computers and Programs
 - *Into. to Computer Science*
 - *Computer hardware basics*
 - *Programming languages and Python 3*
 - *Compiler vs Interpreter*
- Variables and simple loops
 - *Assignment statements*
 - *Data types*
 - *Numerical and logical operators*
 - *Limits to numerical data*
 - *Simple loops and examples*

Unit summary

- Software development process
 - *Introduction of Thonny*
 - *Problem analysis*
 - *Design*
 - *Testing / debugging*
 - *Pseudocode*
 - *print function*
- Writing code in Python
 - *Defining and executing functions*
 - *Parameters and arguments*
 - *Creating and importing modules*

Unit summary

- Accumulator
 - *Accumulator concept*
 - *range function*
- Decisions
 - *Comparison and logical operators*
 - *if, if-else, if-elif statements*
 - *Nested if statements*
 - *Examples*

Unit summary

- Functions
 - *Using built-in functions*
 - *Scope of variables*
 - *Functions and input parameters*
 - *Function return by value and by reference*
 - *Default values for parameters*
- Strings
 - *String data type*
 - *String operations*
 - *Indexing, slicing, concatenation and repetition etc.,*
 - *String methods*
 - *String representation*
 - *Encoder-decoder example*

Unit summary

- Lists
 - *Sequences*
 - *Functions related to lists*
 - *Slicing of lists*
 - *Lists operations*
 - *Map and filter*
- File processing
 - *String formatting*
 - *F string*
 - *Multi-line strings*
 - *Reading and writing in files*
 - *Use with statement*

Unit summary

- Loops
 - *Definite loops*
 - for loop
 - *Indefinite loops*
 - while loop
 - *Sentinel loops*
 - *Nested loops*
 - *break-continue statements*
 - *Examples*

Unit summary

- Exceptions
 - *Exceptions handling*
 - *try-except blocks*
 - *Exceptions hierarchy*
- Sequences
 - *Dictionaries*
 - *Tuple*
 - *Set*
 - *Sequence operations*
 - *Examples*

Unit summary

- Simulation design
 - *Analysis and specification*
 - *Pseudo random numbers*
 - *Top-down design*
 - *Bottom-up design*
 - *Unit testing*
 - *End-to-end testing*
 - *Simulation*
 - *Prototyping and spiral development*

Unit summary

- Objects
 - *Object oriented programming*
 - *Classes, objects and instances*
 - *Graphics library and examples*
- Search algorithms
 - *Linear search*
 - *Binary search*
 - *Comparison of linear and binary search*
- Recursion
 - *Recursive problem solving*
 - *Recursion vs iteration*

PLEASE DON'T FORGET
survey

THANK YOU
&
GOOD LUCK IN YOUR
EXAM