FIT9136 – Lab Activity 1

Assessment

Marks As part of this unit, workshops are not assessed. Attendance is not mandatory, however the sheets are designed so that you get the most out of them via class discussions and problem solving along with your tutor.

Expectations

You will not be able to pass this course via passive attendance. Coding is an art, and great artists practise as often as time allows. It is expected that you complete every question sheet each week, and along with this, try to challenge yourself beyond what you can already currently do.

For some this will mean finishing the sheet itself, for others it will mean seeking out greater tasks. In this course there can be a massive skill gap between each student, thus comparing yourself to your fellow classmates is no way to construct a healthy mindset for programming.

Talk to your tutors about how to challenge yourself, its why they are here.

TLDR: Whatever your skill level, we expect you to try and improve yourself.

Useful Material

Python Interpreter: Python is an interpreted language. As such it needs an interpreter to run your code on your machine. Usually people use an interpreter from Python's main website, but for this unit we use Anaconda `https://www.anaconda.com/` This contains a normal Python interpreter with a lot of extras that are universally useful to scientists.

Be sure to download whatever the latest 3.X version of Anaconda is available. Do not download 2.7 as it has become depreciated and thus will be less useful to learn.

Python Documentation: `https://docs.python.org/3/` Coding involves becoming familiar with the dif- ferent facets of your language. Think of the words that you write to perform code as tools. These tools were designed with specific intentions and purposes. You can find the specifics of individual functions and coding facets through this website. [1]

Getting Started: Should you find yourself in need of a coding environment while connected to the internet try Python Tutor `http://pythontutor.com/` Its a useful resource for if you need to code something quickly, but more importantly is that it visualises the running of the code. If you are struggling to visualise what is happening in your code, this is your first port of call.

You'll need an IDE: There are many Integrated Development Environments (IDEs), but for this unit we recommend Pycharm `https://www.jetbrains.com/pycharm/` Be sure to download this before your first class. It is recommended that you install Pycharm AFTER you install Anaconda. Also be sure that you download the Community Edition. Its free and open source.

You can also find this documentation in python itself by using help()

Before you arrive

Setting up Python on your own machine

You are almost guaranteed to need to install Python on your own machine. Try to do this before your first class. Whatever device your choose to use, please ensure that it has a keyboard. Do not try to flex by going through this course exclusively on your phone. It will end in tears. Please see Useful Material (above) on where to get the Python interpreter for this unit and an IDE to write code with.

Task 1: Today's Algorithms

Part A - What have you done today?

List all the algorithms you have performed today. If you are doing this early in the morning, try to list the algorithms from yesterday as well.

Part B - Condensing Algorithms

From your list, try and categorize the things you have done into groups of similar/same algorithms. For example, if you cooked breakfast and cooked lunch, these would be the same category of algorithm (if not the exact same algorithm).

Part C - Elimination

Recall the features of an algorithm from your lecture. Using these, justify if each of your listed tasks are actually algorithms or if they are something else. Remember that sometimes the input to an algorithm is just starting it (e.g. looking for your keys).

Task 2: Examining Algorithms

Through this course you will spend most of your time determine algorithms to solve problems posed to you. However one aspect of this is knowing if what you have come up with is considered an algorithm. If you are finding it difficult to justify your solution as an algoirthm, you will struggle to program it.

Which of the following are algorithms, and why:

The dream

```
Put together a crew worthy of an Ocean’s 11
film Obtain the secret herbs and spices blend
Bootleg chicken through Peri Peri Start your
own university using the profits Graduate top
of your class
```

Ultimate Study Strategy

```
While there is a book you have not
read:
       Read a book
```

Solve any centralised maze

```
Place hand on left wall Walk through maze, maintaining
hand contact with left wall Stop when the center is
reached
```

Task 3: Primes (Week 2 submission)

Prime numbers are those that cannot be decomposed into factors beyond itself and 1. Such numbers are quite useful for certain programming problems. Thus it falls to you to determine a way of validating a prime number. Using whatever method you wish, describe an algorithm to validate that 7 is a prime.

You may wish to compare with another person’s method. Consider how long your algorithm takes and at- tempt to validate a much larger prime (such as 113).

Task 4: Pebbles

A man comes up to you and suggests you should play his game. The man has a box containing dark and light pebbles. He will reach into the box and pull out two pebbles. If the pebbles match, he will throw them away and put a dark pebble back in. If the pebbles do not match, he will discard the dark pebble and return the light pebble.

He makes you an offer: bet some money, and if you guess what the last pebble will be, he’ll double your money. Should you play the game?

Task 5: The Wolf, The Rabbit, and The Cabbage (week 2 submission)

A farmer must move his wolf, rabbit, and cabbage to the other side of a river. The problem is that if he leaves the wolf alone with the rabbit, the wolf will eat the rabbit. If he leaves the rabbit alone with the cabbage, then the rabbit will eat the cabbage.

The farmer’s boat can only carry himself and one other object at a time. What series of steps are needed to cross the river?

Important

You have to submit Task 3, 5 in Lab activity 1 AND Task 8, 9 in Lab activity 2 to moodle week 2 submission. There is no late submission allowed.