Week 1 Practical: Getting Started

Overview

Due to this practical potentially occurring prior to any lecture this practical will revise concepts previously covered to ensure students have a basic start. This week's practical will introduce students to python (to ensure everyone has the basics), discuss ethical issues in Data Mining and Machine Learning, and practice basic research skills.

Practical Learning Outcomes

After completing this practical students will be able to:

PLO1 – setup python projects that have a framework for loading, preparing, processing and reporting on Data mining problems using machine learning algorithms

PLO2 – find and recognise quality academic papers using google scholar.

Class Activity

You will work on this activity either individually, in a small group or with the whole class as directed. This part of the practical does not need to be submitted, but will provide the formative learning experience that will help you to do the Individual Problem solving task, which is required to be submitted.

Task 1: Problem Solving in Python

In this task you will solve some simple problems using Python. The first three subtasks are primarily revision and can be skipped if you are confident. If you skip them and then find you have difficulty doing subtask 4 then you should go back and do the relevant skipped subtasks.

Students new to programming, new to Python, or have simply forgotten should do the first three subtasks.

Subtask 1: Introduction to Python Environments

If you haven't used Python before or have forgotten how to use it then do this subtask. Otherwise, if you are familiar with Python then you can skip subtask 1.

Python (http://www.python.org/) is a modern, general-purpose, object-oriented, high-level programming language. We will be using Python 3.6 in this unit. Specifically we will be using the Anaconda distribution.

To install and get started with Python work through the "1. Introduction to Python Environments.pdf".

Subtask 2: Introduction to Jupyter Notebook

If you haven't used Jupyter Notebook before or have forgotten how to use it then do this subtask. Otherwise, if you are familiar with Jupyter Notebook then you can skip subtask 2.

To get started with Jupyter Notebook work through the "2. Jupyter Notebook Introduction.pdf" based on material from the pre-requisite unit SIT112.

Subtask 3: Python Programming Revision

If you are not a confident Python programmer then you should do this subtask. Otherwise, if are confident then you can skip this subtask. Note you may need to return to this subtask if you find the next subtask is too hard.

The file "3. Introduction to Python Programming.pdf" contains a number of Python programming references and basic formative learning exercises. After working through these you should have a basic level of Python programing skills.

Note: If you have never done programming before you should further investigate each type of operation discussed to ensure you understand. This review is really only suitable for students that already understand basic programming.

Individual Problem Solving Task

Your individually worked solution to this section should be submitted as part of your Assignment 1 Problem Solving Task Part A (Weeks 1 -3).

Task 2: Problem Solving in Python

Work through "6. Problem Solving Tasks.ipynb" and submit your worked solutions.

If you can't do this return to the subtasks for Task 1.

Extension Work

This section of the practical is not compulsory and not assessed. You do not have to do this and it will not help you gain a higher grade in this unit. The purpose is to challenge students that want to go beyond the unit.

Problem: (adapted from the ACM-ICPC Programming Competition World Finals https://icpc.baylor.edu/worldfinals/problems 1999)

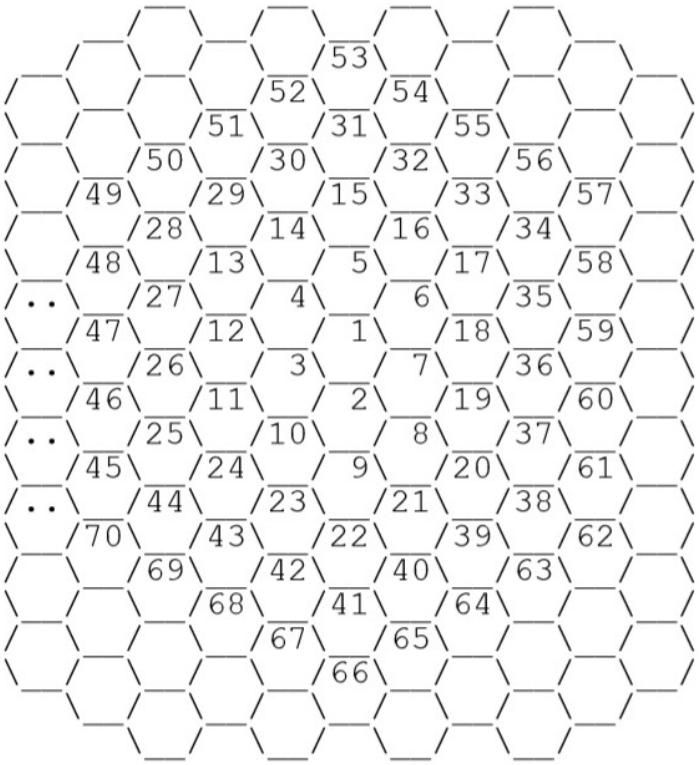
Solve the following using Python – no solution will be provided.

Professor B. Heif is conducting experiments with a species of South American bees that he found during an expedition to the Brazilian rain forest. The honey produced by these bees is of superior quality compared to the honey from European and North American honey bees. Unfortunately, the bees do not breed well in captivity. Professor Heif thinks the reason is that the placement of the different maggots (for workers, queens, etc.) within the honeycomb depends on environmental conditions, which are different in his laboratory and the rain forest.

As a first step to validate his theory, Professor Heif wants to quantify the difference in maggot placement. For this he measures the distance between the cells of the comb into which the maggots are placed. To this end, the professor has labeled the cells by marking an arbitrary cell as number 1, and then labeling the remaining cells in a clockwise fashion, as shown in the following figure.

For example, two maggots in cells 19 and 30 are 5 cells apart. One of the shortest paths connecting the two cells is via the cells 19 - 7 - 6 - 5 - 15 - 30, so you must move five times to adjacent cells to get from 19 to 30.

Professor Heif needs your help to write a program that computes the distance, defined as the number of cells in a shortest path, between any pair of cells.



Input

The input consists of several lines, each containing two integers D and E (D,E ≤ 10000), denoting numbers of cells. The integers are always positive, except in the last line where D=E=0 holds. This last line terminates the input and should not be processed.

Output:

For each pair of numbers (D,E) in the input file, output the distance between the cells labeled D and E. The distance is the minimum number of moves to get from D to E.

Sample Input	Sample Output
19 30	The Distance between cells 19 and 30 is 5.
81 97	The Distance between cells 81 and 97 is 11.
0 0	