# Learner resource 1.1

**Activity 1: Divide and conquer.**

Task A: How does Quicksort use the so-called divide and conquer strategy?

Task B: Using RosettaCode.org site find Quicksort implementation in your favoured language. Copy it out and supply with annotations that explain what each line does and especially how recursion makes it possible. (http://rosettacode.org/wiki/Sorting_algorithms/Quicksort, available under GNU Free Documentation License 1.2.)

**Activity 2: Backtracking.**

Task A: After looking at the definition of computational 'backtracking', put the definition in plain English terms and identify a list of three real-life (non-computing) applications of backtracking.

Task B: Which computational problems can be assisted through visualisation?

**Activity 3: Visualising data and data mining.**

Pilot users were asked to rate the new interface for the upcoming update of the office software which was meant to address the complaints that the previous version was confusing the novices. The users who participated in the study were asked two questions:

'On the scale of 1–10 (where 1 is unusable and 10 is fully usable without training or reading the manual) rate the ease of use of the interface'

'On the scale of 1–10 (where 1 is not comfortable with computers at all and 10 is expert) rate the level of your general computer knowledge.'

# Learner resource 1.1

The software supplier is trying to get a sense of how the level of experience correlates with the perception of the new interface.

The responses were as follows:

| Name | Rating of Interface | Level of Computer Experience |
|------|---------------------|------------------------------|
| Jack | 2 | 1 |
| Bill | 6 | 5 |
| Norma | 8 | 6 |
| Gulchita | 9 | 8 |
| Aron | 5 | 6 |
| Amir | 1 | 2 |

# Learner resource 1.1

Task: Develop a program that plots the responses, so that a visual correlation can be established (correlation is stronger when data points are arranged in a narrow cloud and is weaker when that cloud is rounder). You can use this diagram as a guide:
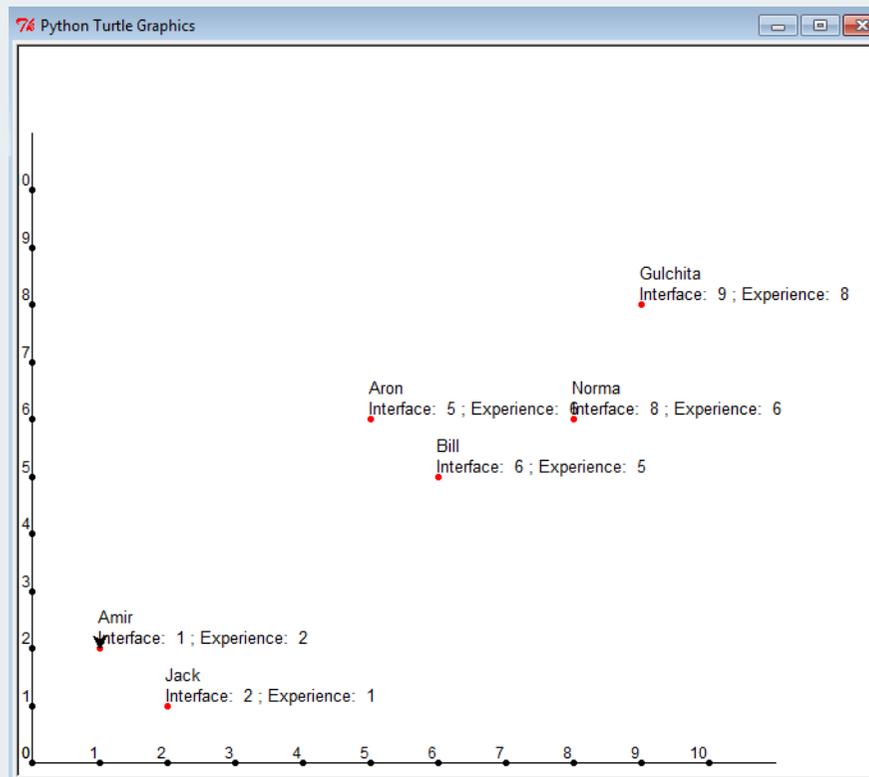


**Activity 4: Shuffling a list.**

Task: Given this list, how would you shuffle it to randomise the position of the elements? (Elements must be unchanged.)

ar = [5,2,8,9,5,7,3]