# Carry1st Data Scientist - Technical Assessment

The technical assessment is designed to test the candidate’s problem solving skills and the ability to frame a problem the correct way.

1. ML Assessment (Verbal)
2. Algorithm Assessment (Python)

The ML assessment will be performed verbally with Suresh - VP Data. It will be a high to mid-level discussion. Model/Code/implementation specifics will not be assessed. Rather, how you understand and frame the problem, along with the proposed solution, will be considered

For the Algorithm assessment, please provide full documentation - basic explanation of solution and the result, Python code that executes without error, test cases.

## 

## ML Assessment

Multi-touch marketing attribution is on the most challenging topics in Data Science, due to its ambiguity.

Consider the case of an e-Commerce website like eBay or Amazon. A user/customer may land on the site either directly (organically) or via marketing channels (Google search results, Good paid search ad, Affiliates, Banner ads, FB ad, ….). Organic can be considered a channel.

When a conversion is made, we need to attribute some portion of the value of the purchase back to each marketing touch point, to reflect their influence on driving the conversion.

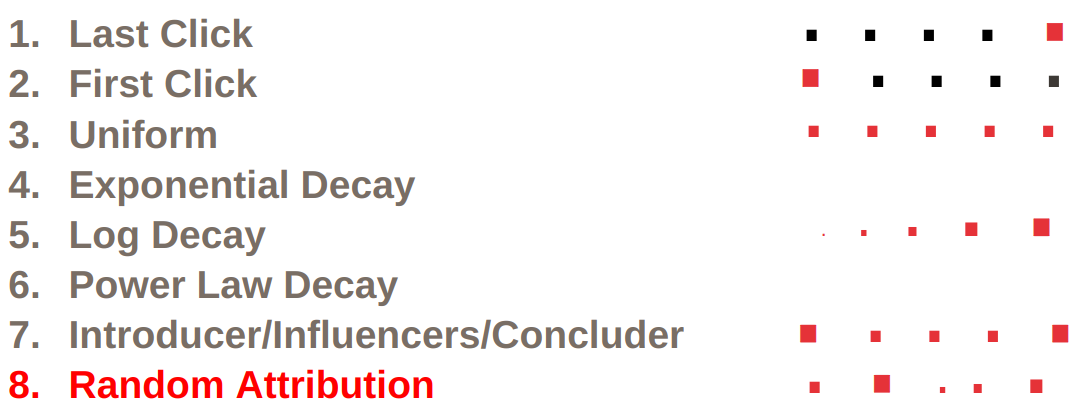
| ⯀ | ⯀ |  | ⯀ |  | ⯀ | ★ |
| --- | --- | --- | --- | --- | --- | --- |
| Channel A | Channel B |  | Channel A |  | Channel C | Conversion |

🠚 time

The goal is to develop a model to perform this attribution. You have access to all external (marketing touch points) and internal (behaviour on the website) data.

| **Be prepared to discuss how to frame the problem and what would be a suitable model to perform the retrospective multi-touch attribution.**  **Some research on the general problem will help the conversation, but a detailed review is not required.**  **Clue: Toy models as pictured below are wrong, and equally so. :)** |
| --- |

[continued on next page …]



## 

## 

## 

## 

## 

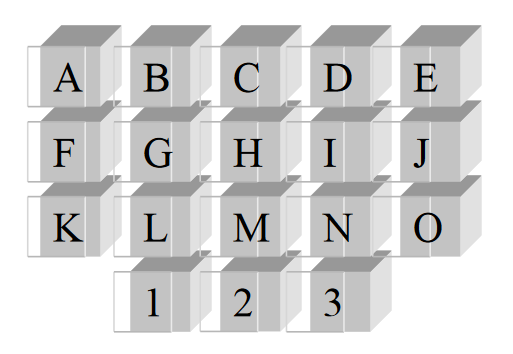
## 

## 

## 

## Algorithm Assessment

**Knight Sequences** – Given the following picture below of a keypad:



Write a python program to find all 10-key sequences that can be keyed into the keypad  
in the following manner:

* The initial keypress can be any of the keys.
* Each subsequent keypress must be a knight move from the previous keypress.
* There can be at most 2 vowels in the sequence.

A knight move is made in one of the following ways:

1. Move two steps horizontally and one step vertically
2. Move two steps vertically and one step horizontally

There is no wrapping allowed on a knight move.

| **Your program should write the number of valid 10-key sequences on a single line to standard out.**  **Both top-down and bottom-up solutions are possible, so please rationalise your choice.** |
| --- |

[continued on next page …]

Below are some examples of knight moves: