*TiltText: Using Tilt for Text Input into Mobile Phones*

* Removes ambiguity from the standard Multi-Tap system in the 12 key text input by using the orientation of the phone- by tilting- to determine which character on a particular key should be selected
  + For example, pressing the key marked as '2' and then tilting the phone left will present an 'a' character whereas tilting the phone forward will present a 'b' character
* TiltText was 23% faster than Multi-Tap
* 3 main techniques in reducing ambiguity as all of the 26 characters must be represented by typical keypad of 9 buttons.
  + Each number is assigned with a selection of letters, in addition to the number the button represents. For example, the '1' button will represent 'a', 'b', 'c' and '1'
    - Multi-Tap
      * To access letter 'c' in button '1', the user must press '1' 3 times for the 'c' to appear
      * The cursor will move along when a new button is pressed. However problems arise when 2 characters are required from the same button. This is resolved by a short timer where no buttons have been pressed
    - Two-Key
      * This is where 2 buttons are pressed within quick succession
        + First button to decide which group of characters
        + Second to decide which position the character you choose is within
    - Linguistic
      * This is where numbers are pressed where the character lies within. Meaning that the system renders words which can be formed from the different categories in the specific order
* Possible design issues occur when the user requires upper-case letters. This is resolved in 1 way as the device can be tilted twice in the same direction, implicating a 'double letter'. Also spaces must also be assigned to a button or unambiguous entry method
* 3 main ways to determine when to calculate the tilt of the phone when a button is pressed
  + Key Tilt
    - This involves calculating the tilt of the phone between the duration that the button is pressed down to when the button is released
  + Absolute Tilt
    - This involves calculating the tilt of the phone just before the button is pressed. The tilt of the phone must lie on the horizontal plane before beginning
  + Relative Tilt
    - This involves calculating the change of tilt of the phone between each of the button presses. For example 30 degrees left will show 'a' and a 15 degrees right will show 'c'
* Evaluation