

Computer Programming 4

Object Oriented Programming

CA269

Assignment Outline

Background



Before the advent of QWERTY keyboards, texts and numbers were placed on the same key.

For example, 2 has “ABC” if we wanted to write anything starting with ‘A’ we need to type key 2 once. If we wanted to type ‘B’, press key 2 twice and thrice for typing ‘C’. Below is a picture of such a keypad.



Given a keypad as shown in the diagram, and an n digit number provided by user,

Working in groups of 4,

Write a Java program to list (in alphabetical order) all letter combinations which are possible by pressing these numbers.

For example, if the **Input number:** 234

The Output should be:

adg, adh adi aeg aeh aei afg afh afi bdg bdh bdi beg beh bei bfg bfh bfi
cdg cdh cdi ceg ceh cei cfg cfh cfi

i.e., if 2 is pressed then the alphabet can be a, b, c,
Similarly, for 3, it can be d, e, f,
for 4 can be g, h, i.



Suggested Approach

- **Approach:** It can be observed that each digit can represent 3 to 4 different alphabets (apart from 0 and 1).
- So you could form a recursive function. Then map the number with its string of probable alphabets, i.e 2 with “abc”, 3 with “def” etc.
- The recursive function will try all the alphabets, mapped to the current digit in alphabetic order, and again call the recursive function for the next digit and will pass on the current output string.

My approach would be to...

- Map the number with its string of probable alphabets, i.e., 2 with “abc”, 3 with “def” etc.
- Create a recursive function which takes the following parameters, output string, number array, current index, and length of number array
- If the current index is equal to the length of the number array, then print the output string.
- Extract the string at *digit[current_index]* from the Map, where the digit is the input number array.
- Run a loop to traverse the string from start to end
- For every index again call the recursive function with the output string concatenated with the character of the string and the *current_index + 1*.

Marking

- Complete Working Java Program and ReadMe file 10%
 - ReadMe file will include group member names, student number, breakdown of work
- Creative element 5%
 - Each group will give something different
 - For example ask user to enter their name – then tell them which keys they need to press to spell their name

Submission

- Due End Week 6 20th February @ 11.59pm
- Email Assignment files to tracey.mehigan@dcu.ie

Please Comment Group details at start of your java program

```
/* -----  
Group Member Names/Student Numbers  
*/
```