

University of Massachusetts Dartmouth
Department of Electrical and Computer Engineering

ECE 161
Lab 4

Due: <http://ece161.viall.org>
Name: buildcombo.cpp

The objective of this lab is to write a subsection of Lab 5. Lab 5 will be to write the function:

```
int buildwords(char tiles[], int numtiles, char words[][13]);
```

Where:

tile[] – list of the available letters. A max of 7 letters are available.

numtile – the number of tiles that are available.

words[][13] – this is the list of letter combinations

w - will be the global array of all dictionary words as used in isword()

Lab 4 is involved in constructing all of the letter combinations that can be made using the letters in tile[].

Logic:

Consider a small subset of the problem. Let's say we have the letters T, H, A, and N. The various combinations that we can have are as follows:

```
THAN, THNA, TAHN, TANH, TNHA, TNAH, HTAN, HTNA,  
HATN, HANT, HNTA, HNAT, ATHN, ATNH, AHTN, AHNT,  
ANTH, ANHT, NTHA, NTAH, NHTA, NHAT, NATH, NAHT,
```

(note there will be n! combinations where n is number of tiles)

If I have an array called tiles[], where

```
tile [0]='T'; tile [1]='H'; tile [2]='A'; tile [3]='N';
```

then what I really need to do is generate combinations of 0, 1, 2, 3 and plug the generated numbers into the subscript of tiles[].

For example, consider the code:

```
for (i=0; i<4; i++)  
{  
    for (j=0; j<4; j++)  
    {  
        for (k=0; k<4; k++)  
        {  
            for (m=0; m<4; m++)  
            {  
                printf("%1d%1d%1d%1d\n", i, j, k, m);  
                // or printf("%c%c%c%c\n", tile[0], tile[1], tile[2], tile[3])  
            }  
        }  
    }  
}
```

The problem with this code is it produces duplicates of numbers (0000,0001, etc). To alleviate this problem, consider putting some type of test just inside each of the j, k, and m loops (akin to `if (i==j)...`).

OK, so what is Lab 4 supposed to do?

Here's a sample run:

Enter 4 letters: wxyz

wxyz, wxzy, wyxz, wyzx, wzxy, wzyx, xwyz, xwzy, xywz, xyzw, xzwy,
xzyw, ywxz, ywzx, yxwz, yxzw, yzwx, yzwx, zwxy, zwyx, zxwy, zxyw,
zywx, zyxw

Enter 4 letters: zyxw

zyxw, zywx, zxyw, zxwy, zwyx, zwxy, yzwx, yzwx, yxzw, yxwz, ywzx,
ywxz, xzyw, xzwy, xyzw, xywz, xwzy, xwyz, wzyx, wzxy, wyzx, wyxz,
wxzy, wxyz,

Enter 4 letters: *