# Question 1: Part D Automatic Testing

Pardon for not coding a unit test for the program, as I do not how. But to test the viability of the code, one considers the coverage of inputs.

## Informal testing

Assume a user puts a single character and ‘a’ is a word in English but just like that 27 unique combinations can be inserted, for this test case scenario, a test code of any single character can be used to fulfill the whole spectrum of single inputs.

Increasing the alphabets complicates things, as the variation will increase, take 2, and 3 alphabets as an input, then words can be formed from the following shown in the below table.

However, after reaching the 4th alphabet onwards basically iterations or any sum of odd or even characters for input. Using, Equation 1.0 and 2.0 are used to determine how many 1st and last characters to be placed and reverse order of the 1st set respectively, then any set of unique or the same characters is to be placed in the middle.

|  |  |  |  |
| --- | --- | --- | --- |
| Test  case | No. of alphabet(s), N. | Description | Example |
| 1 | 1 | Any Random Character | a |
| 2 | 2 | Any Random Character | ab |
| 3 | All Same Character | aa |
| 4 | 3 | Any Random Character | abc |
| 5 | All Same Character | aaa |
| 6 | 1st and Last Character same but unique to middle character | aba |
| 7 | 4  (even nth characters) | Any Random Character | abcd |
| 8 | All Same Character | aaaa |
| 9 | 1st and Last (Ce) Characters in palindrome order but unique to the middle characters all unique to each other | abca |
| 10 | 1st and Last (Ce) Characters in palindrome order but unique to the middle characters all the same to each other | ab\*ba |
| 11 | 5  (odd nth characters) | Any Random Character | abcde |
| 12 | All Same Character | aaaaa |
| 13 | 1st and Last (Co) Characters in palindrome order but unique to the middle characters all unique to each other | ab\*ba |
| 14 | 1st and Last (Co) Characters in palindrome order but unique to the middle characters all the same to each other | ab\*ba |
| ~ | Nth  (repetitaive to previous) | Follows the same rules for any even number of characters (4 onwards) and any odd number of characters (5 onwards) | ~ |

## Approach to testing

By implementing a function outside or in the main function, to return all the false and true value, using the test cases above.

Sample:

Header file

bool test()

{

switch(test case)

case (Test case 1):

string reverse array()

strcmp (normal, reversed)

returns true or false

case (Test case 2):

string reverse array()

strcmp (normal, reversed)

returns true or false

case (Test case 3):

.

.

.

.

}