

## Description of logic in flow chart

1. Variables and arrays are all initialized at the start of the flow chart with the exception being "Letters 1-99". Since letters 1-99 has a large amount of letters in it, they will all be initialized at the end of this README file.
2. The "random\_num\_gen = 9?" statement will continue to loop until nine random letters are generated.
3. The letters will then be shown on screen and the user will be prompted to make a word from them.
4. The program will then check if the word used by the user is in the dictionary that the program uses.
5. If it is, the program will continue, if not the user will be given an error message and the program will be looped back to the random\_num\_gen and nine new numbers will be assigned.
6. The program will check if the letters input by the user are contained in the "letters (1-9)" array.
7. If they are in the array, the program will continue. If the letters are not in the array an appropriate error message will be displayed and the program will loop back to the start.
8. The program will then check if the letters contained in the user\_guess are vowels. It will check each letter through a loop where the counter represents the specific letter we are dealing with in the "user\_guess" array. If true, the letter is a vowel, if false the letter is a consonant.
9. If the decision is true, the round and score will be incremented by one and the user will be told that they earned a point for a vowel.
10. The program will then immediately check if the score\_counter (i.e. the length of the specific word we are dealing with) is greater than the best\_word variable.
11. If this statement is true the best\_word variable will be replaced by the user\_letter variable (The user letter is the current word the user has entered). Whether or not the statement for ("Score\_counter greater than best\_word variable") is true or false, the program will still continue onto step fifteen.
12. If the decision is false, the round will be incremented by one and the score will be incremented by two.

13. The user will then be told that they have earned two points for a consonant.
14. The program will then immediately check if the score\_counter (i.e. the length of the specific word we are dealing with) is greater than the best\_word variable. If true the best\_word variable will be replaced by the user\_letter (The user letter is the current word that the user has entered). Whether or not the statement for ("Score\_counter greater than best\_word variable") is true or false, the program will still continue onto step fifteen.
15. The next parts (steps 16 - 23 ) will ensure that each letter can be used only once.
16. Here the program will check if the item counter (Note: "counter" is equal to the specific item we are dealing with) of user\_guess is equal to the item counter of letters(1-9). For example: Is a=a? Is a=b? etc.
17. If the statement is true the letter will be replaced by a tick, so that it can not be used again from the list.
18. The variable "letter\_checker" will then be set to one and counter2 will be incremented by one.
19. If letter\_checker is equal to one, counter3 will be changed by one. (Counter3 represents where in the user\_guess we are. )
20. Counter2 will be set to zero. (Counter2 represents where in the user\_guess we are).
21. Counter will then be incremented by one.
22. The program will then check if counter3 is greater than one. If it is true that means that the program has replaced the letter we are dealing with in the loop and that we can continue onto the next letter. Go to step twenty-four for the following step if the statement was true.
23. If the statement above is false, the program will loop backwards and check if the next letter is equal to the letter in the user\_guess or for simplicity go back to step sixteen in this README file.
24. The program will then check if the counter is greater than the length of the user guess. If true, this means that we have checked every letter the user has entered as his word and we can therefore continue the program.
25. If false we must loop backwards and check if the next letter is in the letters(1-9) array. (step six).

26. The next step in the system is checking if the variable rounds is equal to five. If it is we can tell the user their best\_word and end the program
27. If the rounds variable is not equal to five then we need to loop back to the random\_num\_gen area where the user has nine numbers randomly generated for them. (Go to step six if unclear).

### The ninety-nine letters are assigned as follows:

12 letters:e

9 letters:a,i (Note this means nine of letter "a" and nine of letter "i")

8 letters:o

6 letters:n,r,t

4 letters:l,s,u,d

3 letters:g

2 letters:b,c,m,p,f,h,v,w,y,k

1 letters:j,x,q,z