

Given the attached 50,000 words dictionary, write a program that can play hangman by choosing a letter based on the current state of the board. A board would initially have a series of blanks, each representing one letter in the word. As letters are guessed, all spaces in the word that match the letter should be replaced with that letter. Any letters that have no matches are put on a list of missed letters. Once the list of missed letters reaches 6, the game is lost. Write a C++ program that takes the word as a command-line argument and shows the board after each step as well as ultimate outcome. The guesses should come from the program, not from user input.

```

_ _ _ _ _ missed:
guess: e
_ _ _ _ _ missed: e ✓
guess: a
_ a _ _ _ a _ missed: e
guess: n
_ a n _ _ a n missed: e
guess: m
_ a n _ m a n missed: e
guess: d
_ a n _ m a n missed: e, d ✓
guess: k
_ a n _ m a n missed: e, d, k ✓
guess: g
_ a n g m a n missed: e, d, k
guess: h
h a n g m a n missed: e, d, k

```

7 Blanks → 7 letters

guess & program

Guess - mark frequent letters first

Hangman Won ✓  
lost

As a final step, you should run all the words in the dictionary as inputs to your program and show us the percentage of the words matched correctly.

Your program will be evaluated on the following metrics:

- What percentage of the words was correctly guessed?
- How fast did the program run?
- How clearly the program was written?

85%  
95%  
99% } words correct

Please have your program generate the following output:

- Number of words tested: 50,000
- Number of words guessed correctly: Minimum 40,000
- Correct Guesses (%): Minimum 80.0%
- Time to run: 60 seconds

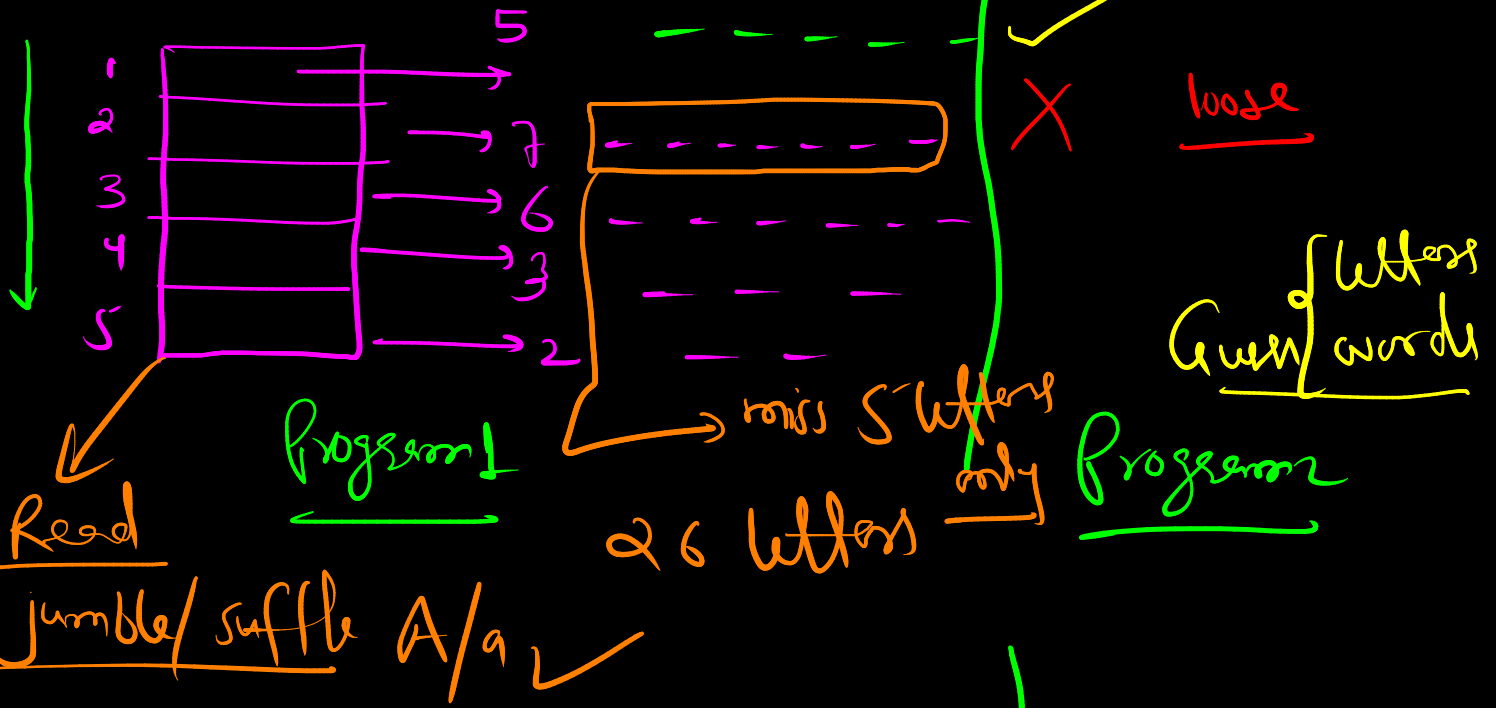
Note - Extend the program to include a GUI (Graphical User Interface), which should be able to browse any other dictionary and run the program accordingly for it. Extend the program further to create the dictionary programmatically that should contain at-least 2,00,000 words. Words in the dictionary should not contain any duplicate.

Program ✓

Python / JavaScript ✓

200000 words

50000 words



string createWord(int n)

return word; length n

3