Hangman

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
```

(Handwritten annotations:) # Blanks → # letters · guess & → program · Guess — most frequent letters first · Hangman Won · lost · XX

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?

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

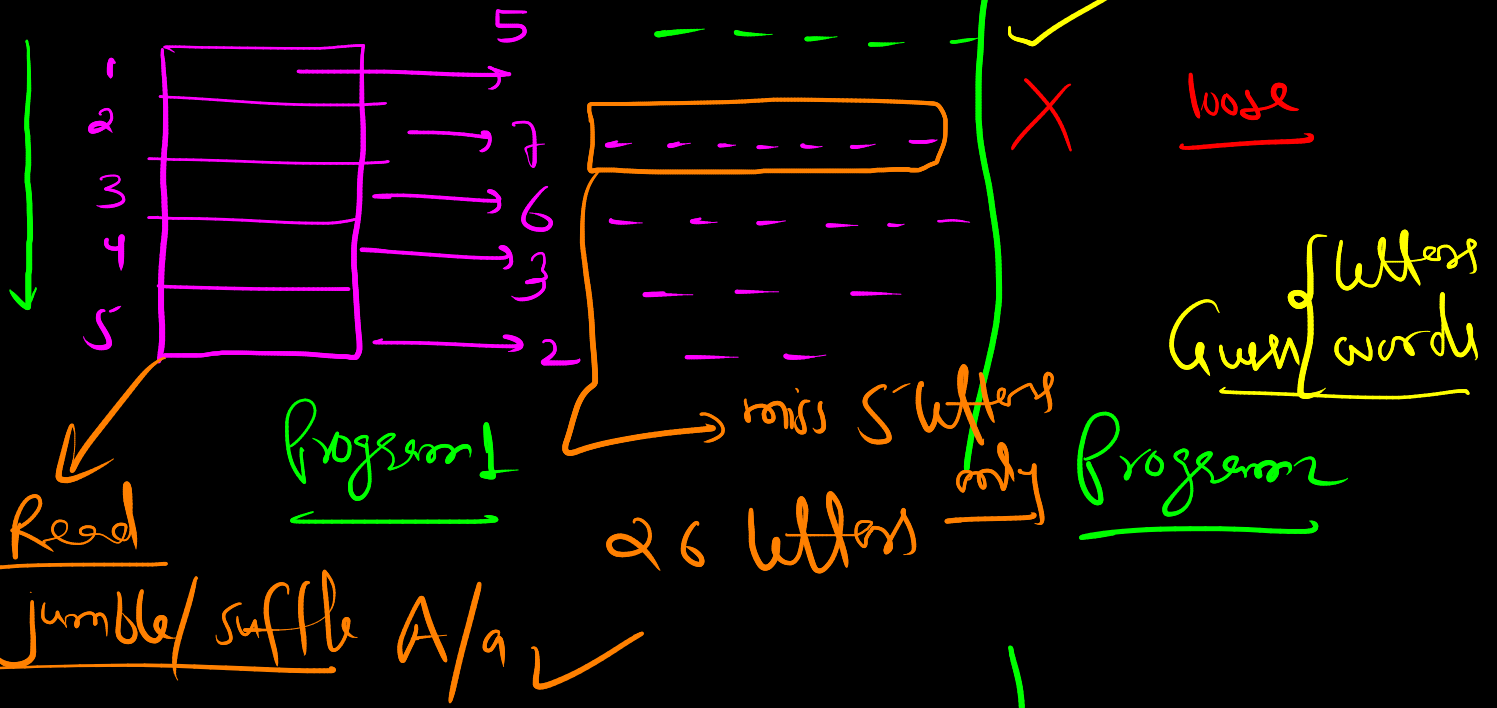*(Handwritten annotations:)* words correct · 85% 95% 99%

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 contain any duplicate.

(Handwritten annotations:) Program · Python / Java Script · 200000 words

50000 words

5

Win

loose

1
2
3
4
5

5 →
→ 7
→ 6
→ 3
→ 2

X

Program1

→ miss 5 letters

only Program2

Read

26 letters

{ letters
Guess { words

jumble/ shuffle A/a

string create word (int n)

return word; length n

&

3