



# WORD SEARCH IN FLOWCHARTS AND PSEUDOCODE



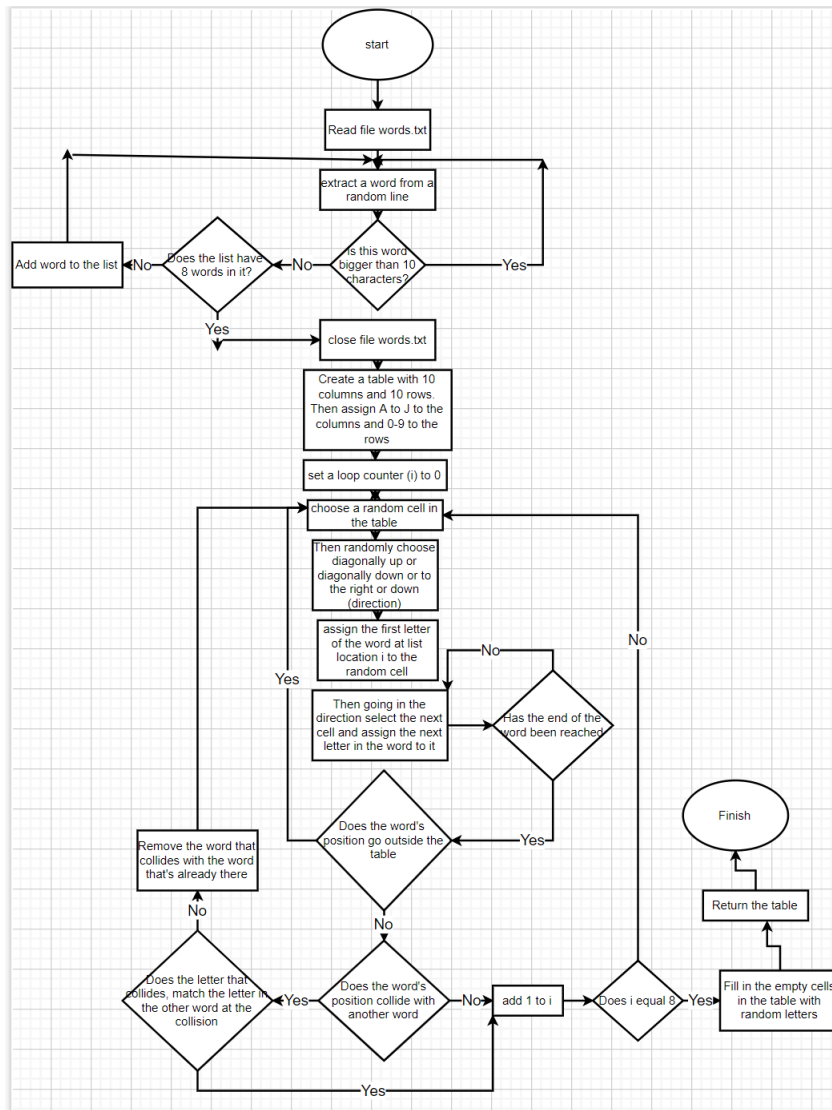
Hugo Hewitt

## Contents

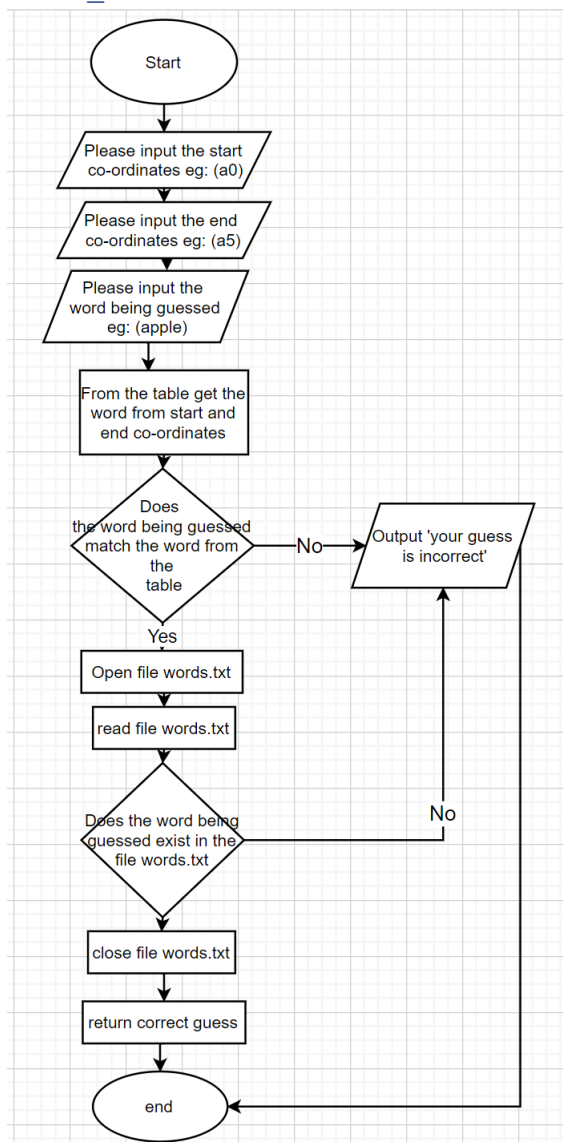
Flow charts.....	2
create_word_search.....	2
check_answer.....	3
main (version 1).....	4
main.....	5
Pseudocode.....	6
create_word_search.....	6
check_answer.....	7
main (version 1).....	8
main.....	9

# Flow charts

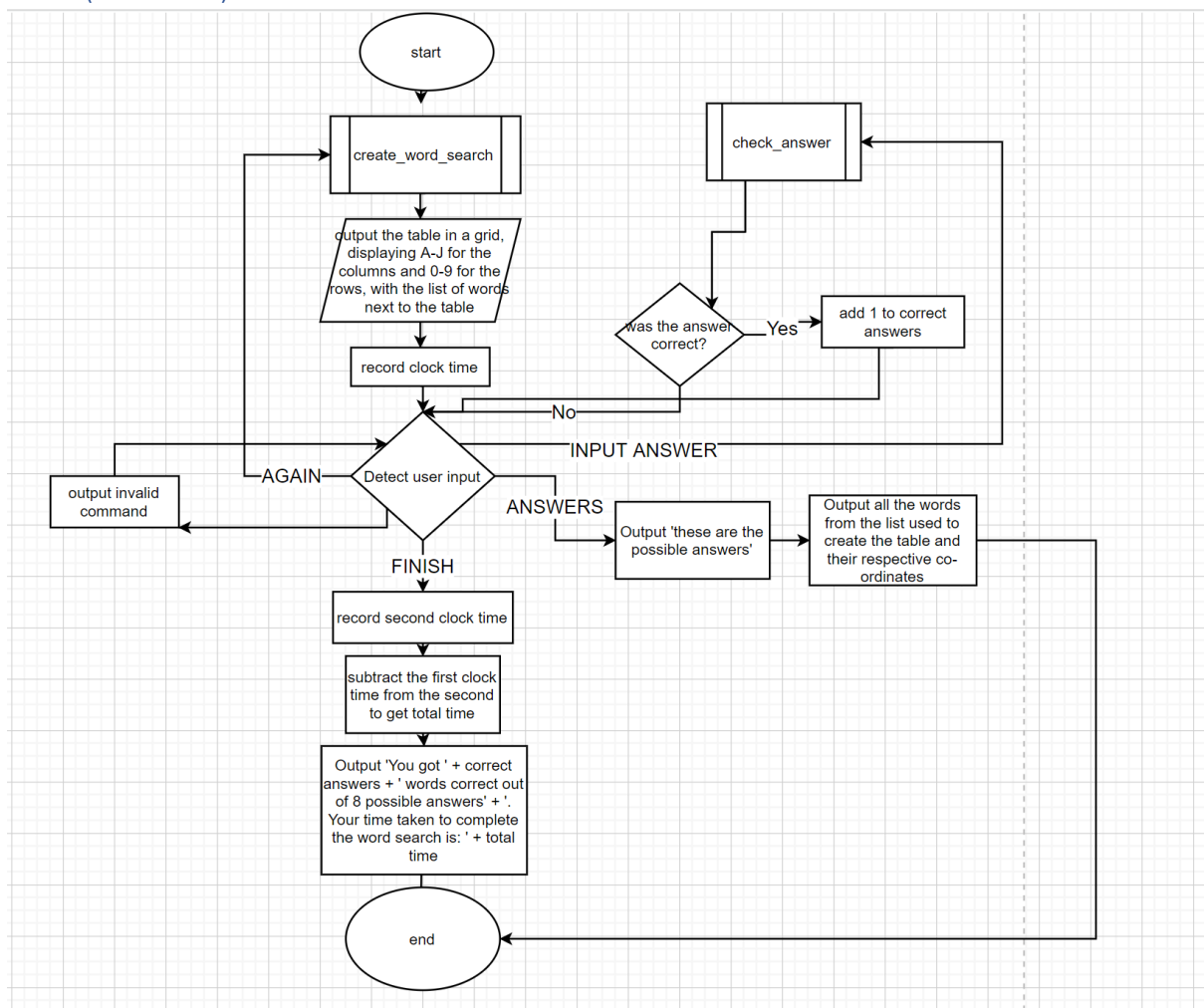
## create\_word\_search



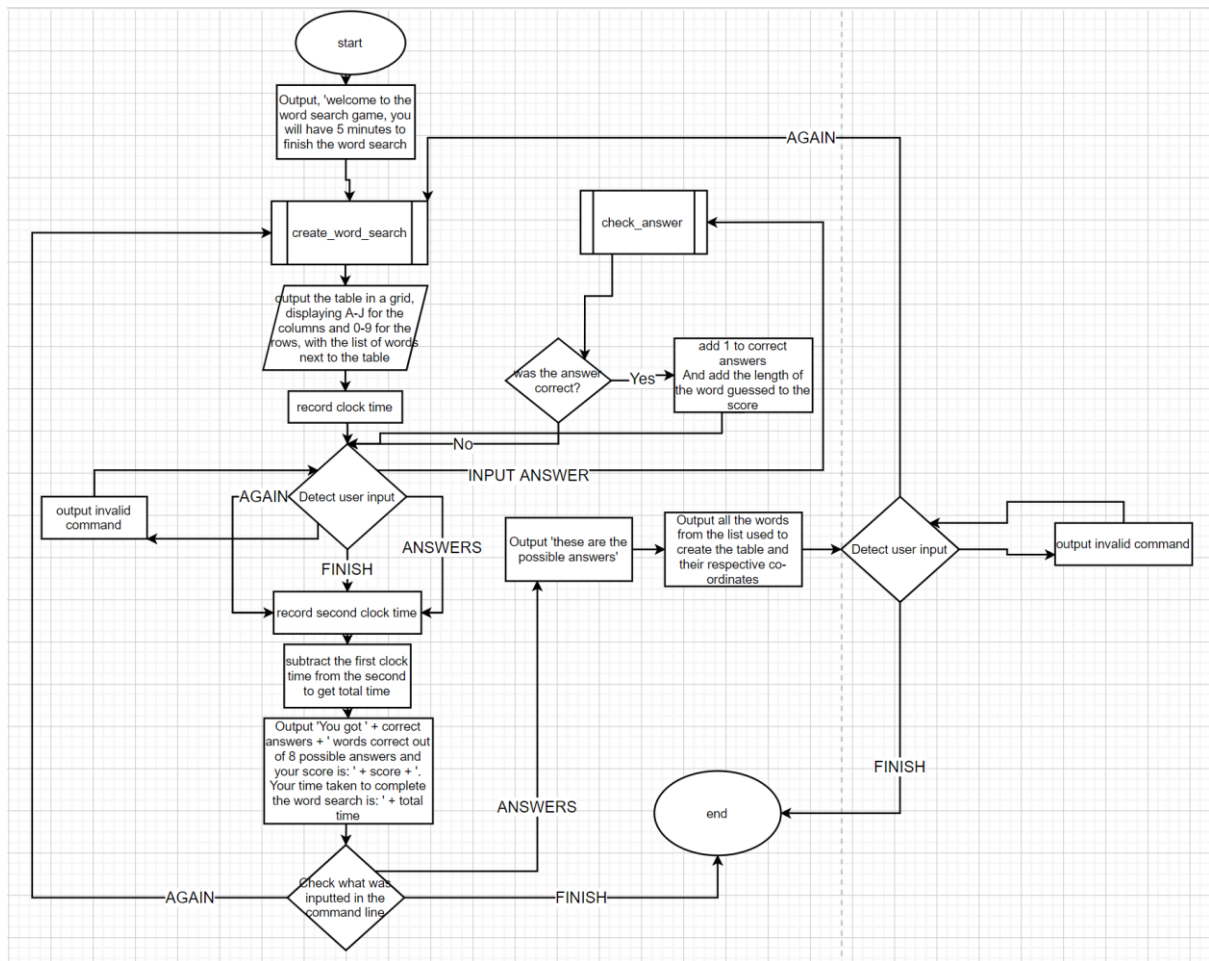
## check\_answer



## main (version 1)



## main



## Pseudocode

### create\_word\_search

Function create\_word\_search:

- Open file words.txt for read

- While there are less than 8 words selected:

  - Select a random word from the file words.txt

  - If selected word is less than or equal to 10 characters:

    - Add selected word to word list

- Close file words.txt

- Generate a table with dimensions 10 cells by 10 cells

- Assign A-J to the columns of the table and 0-9 to the rows

- Assign 0 to a counter (i)

- While there are less than 8 words assigned:

  - Select a random cell in the table

  - Randomly choose a direction from (diagonally up, diagonally down, right, down)

  - Assign the first letter of the word in word list location i to the random cell

  - The next cell is the cell after the random cell in the direction chosen

  - While the end of the word has not been reached:

    - Assign the next letter of the word in word list location i to the next cell

    - Move to the next cell in the direction chosen

  - If the word being assigned goes outside the table:

    - Unassign the current word from the table

  - Else if the word being assigned collides with word already assigned:

    - If the letter that collides matches the letter already assigned:

      - Add 1 to counter (i)

    - Else:

      - Unassign the current word from the table

  - Else:

    - Add 1 to counter (i)

- Fill in the remaining cells in the table with random letters (a-z)

- Return the table

check\_answer

function check\_answer:

ask user to input the start co-ordinates (eg. A0)

ask the user to input the end co-ordinates (eg. A5)

ask the user to input the word being guessed (eg. Apple)

from the table get the word from the start co-ordinates to the end co-ordinates

If word being guessed matches the word from the table:

Open file words.txt

Read all the lines from words.txt

If word being guessed exists in words.txt:

return 'correct guess'

Else:

return 'incorrect guess'

Else:

return 'incorrect guess'

Close file words.txt



main (version 1)

START

generate word search table with function create\_word\_search

display the word search table with A-J on the columns and 0-9 on the rows

record clock time

while word search is not finished

    wait for user input

    if user input is AGAIN:

        go to START

    else if user input is INPUT ANSWER:

        check answers with function check\_answer

        if answer is correct:

            display 'correct answer'

            add 1 to correct answers

    else if user input is ANSWERS:

        display 'these are the possible answers:

        display all words from the list used to create the table and the respective positions

        word search is finished

    else if user input is FINISH:

        record second clock time

        subtract the first clock time from the second clock time to get total time

        display 'your time to finish was: ' + total time + ' You got ' + correct answers +  
        ' correct answers out of a possible 8'

        word search is finished

    else:

        display 'invalid command'

END

main

START

Display 'Welcome to the word search, you will have 5 minutes to finish'

generate word search table with function create\_word\_search

display the formatted word search table with A-J on the columns and 0-9 on the rows

display the list of words used to create the word search

record clock time

while word search is not finished:

    wait for user input

    if user input is INPUT ANSWER:

        check answers with function check\_answer

        if answer is correct:

            display 'correct answer'

            add 1 to correct answers

            if the total time taken by the user is less than or equal to 5 minutes:

                add the length of the word guessed to the users score

            display the formatted word search table and words again

    else if user input is FINISH or AGAIN or ANSWERS:

        record second clock time

        subtract the first clock time from the second clock time to get total time

        display 'your time to finish was: ' + total time + ' You got ' + correct answers +  
        ' correct answers out of a possible 8. ' + 'Your score is: ' + the users score

        if user input was FINISH:

            the word search is finished

        else if user input was AGAIN

            go to the START

        else if user input was ANSWERS

            display 'these are the possible answers:

            display all words from the list used to create the table and their respective  
positions

            Ask the user if they want to do the word search again, yes or no

            If user chooses yes:

Go to start

Else if user chooses no:

The word search is finished

else:

display 'invalid command'

END