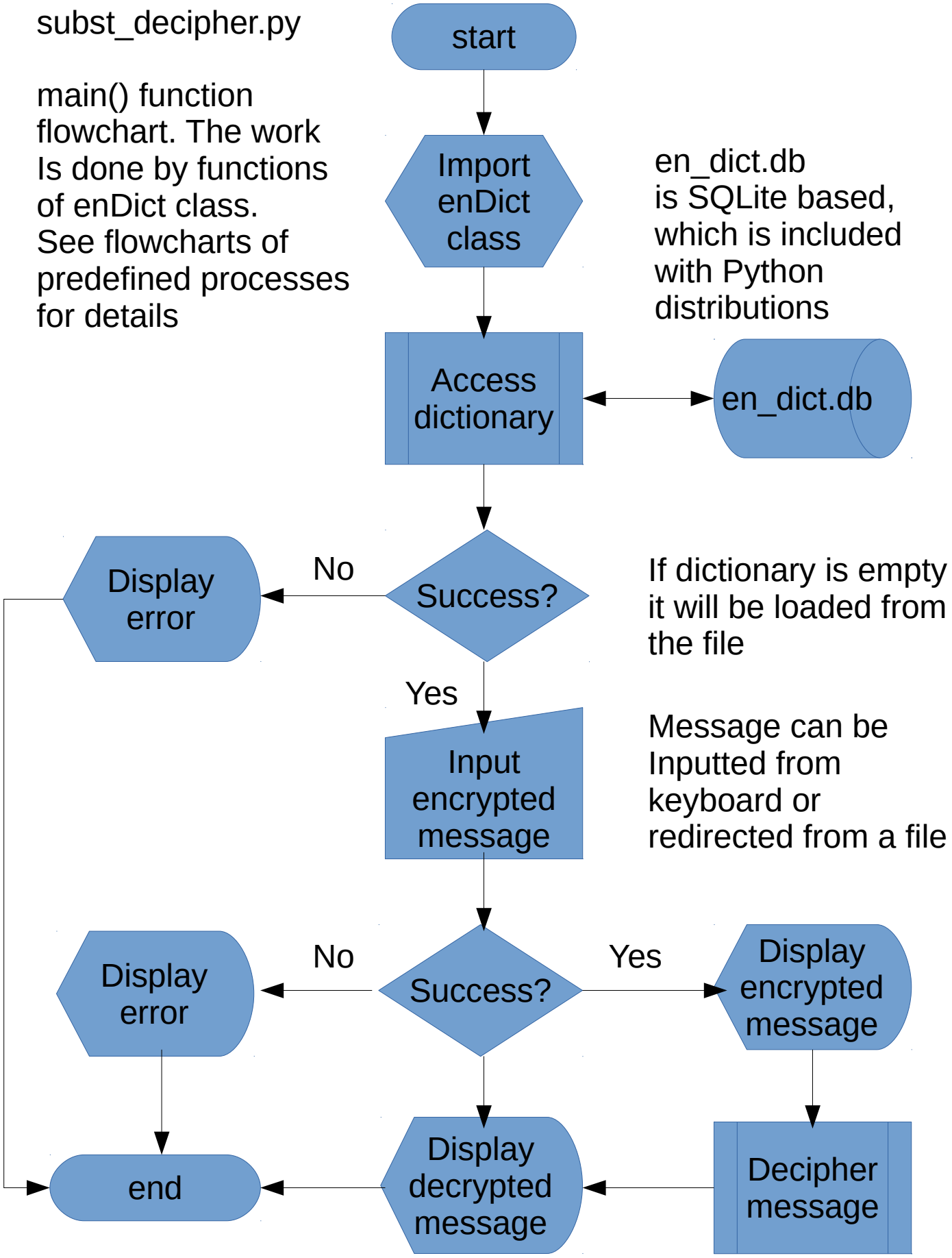
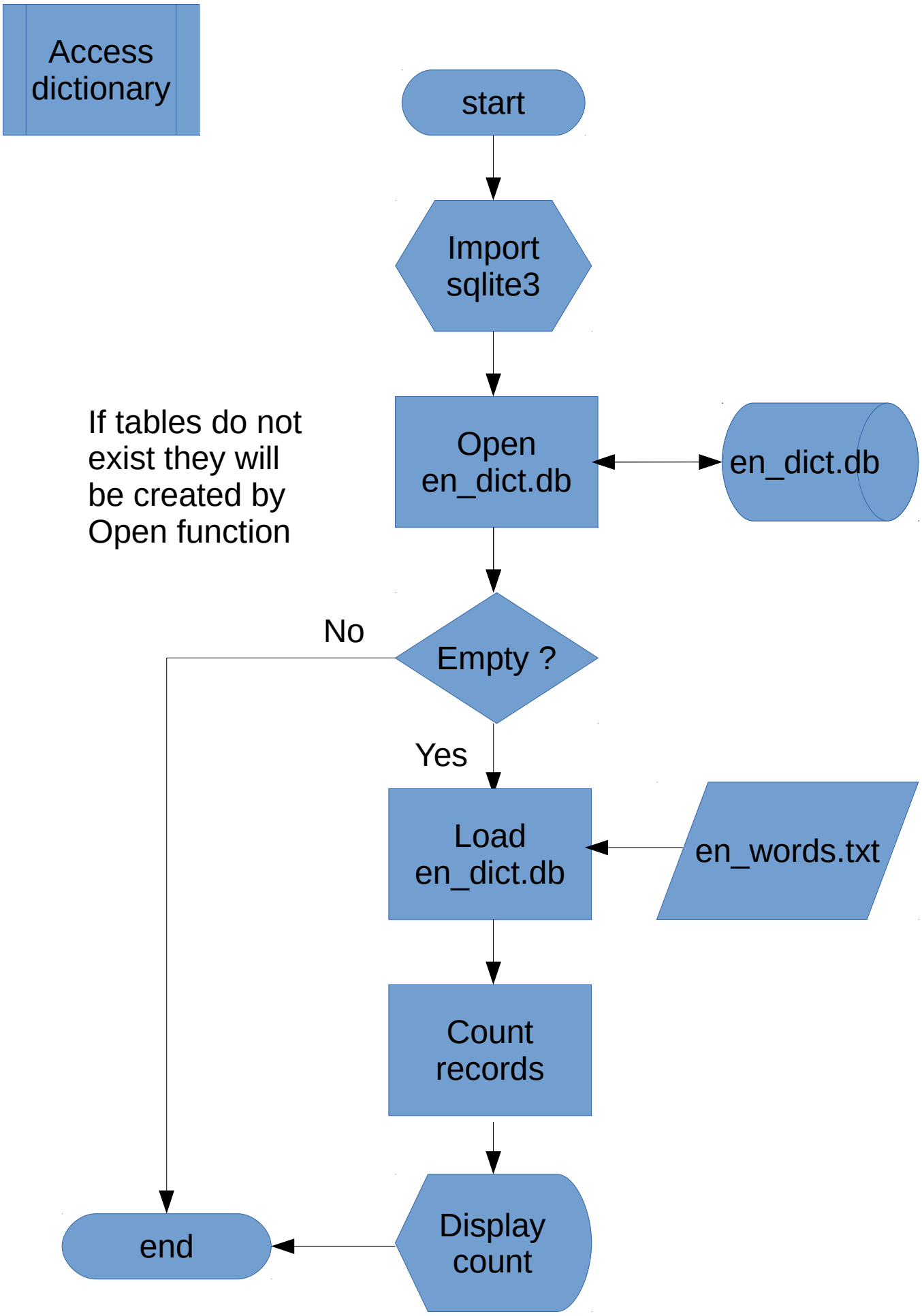
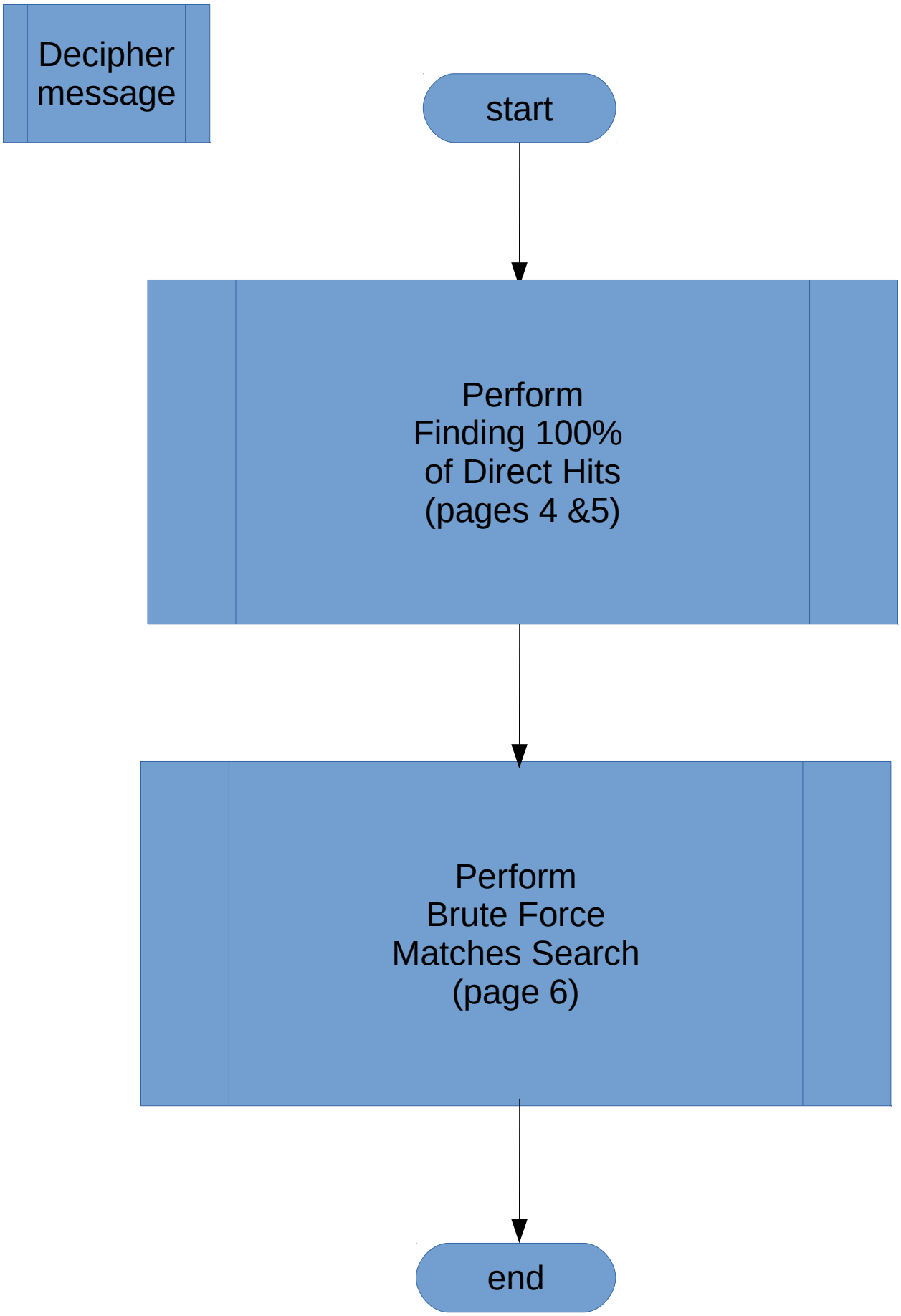
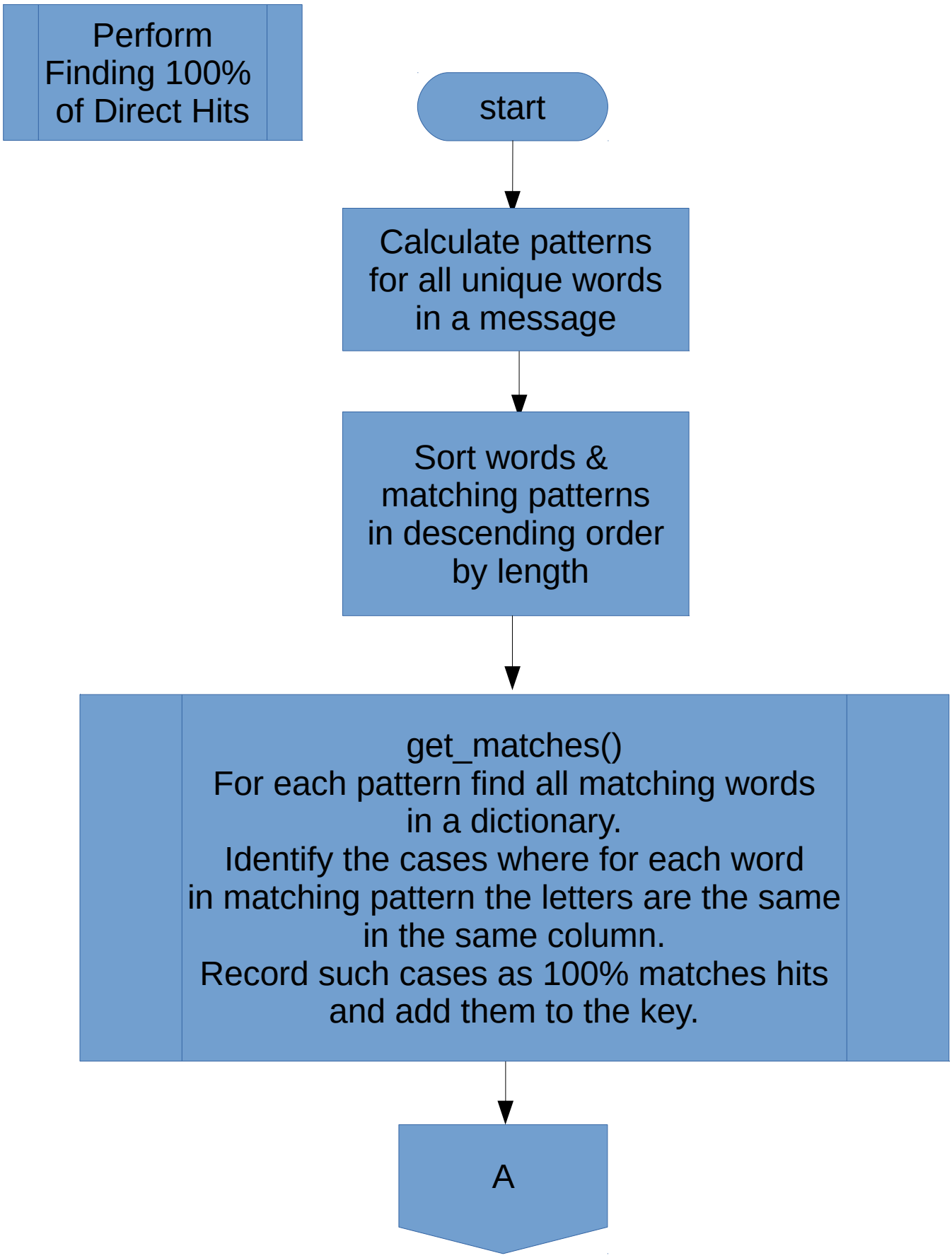
subst_decipher.py

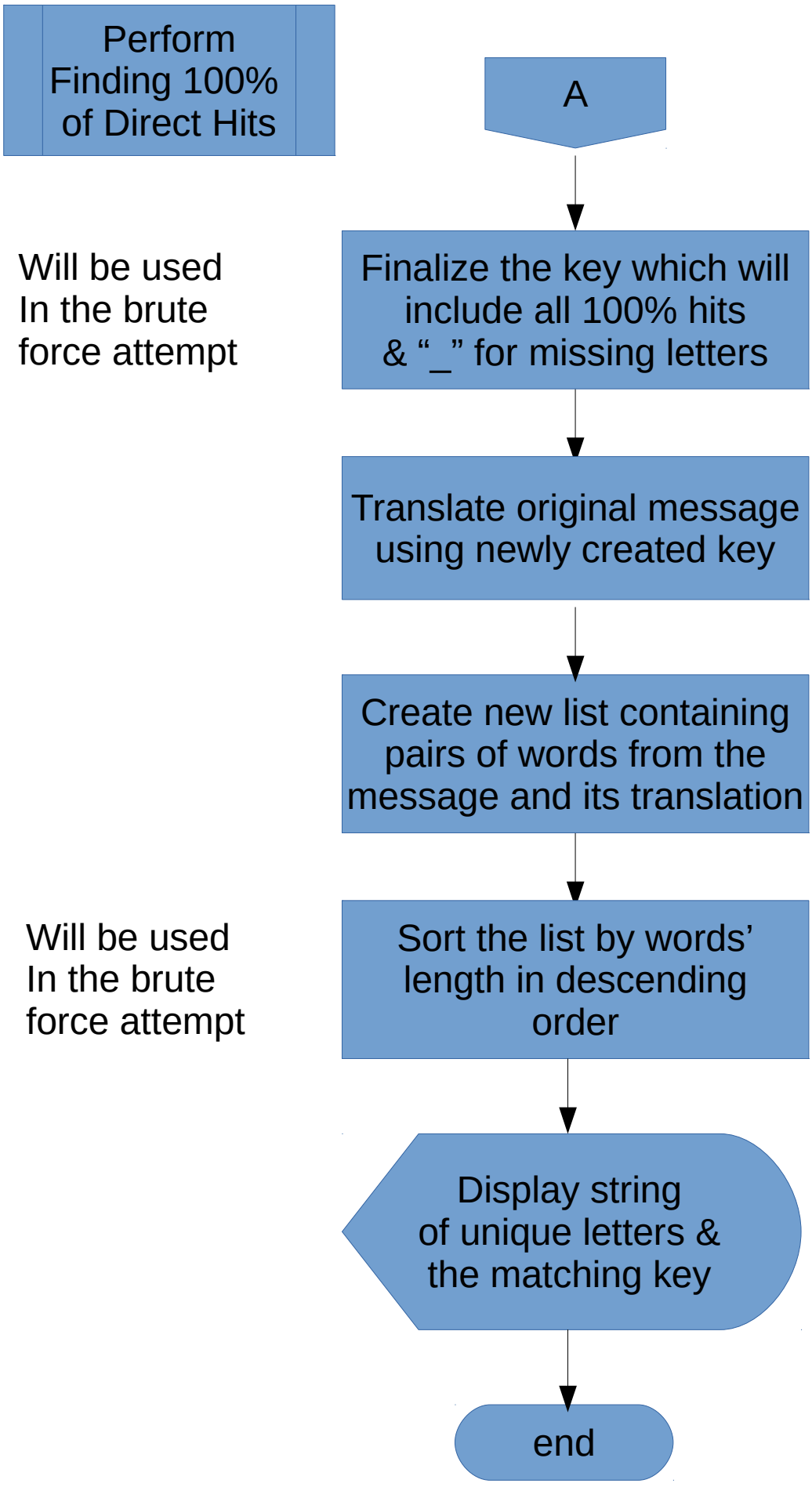
main() function flowchart. The work Is done by functions of enDict class. See flowcharts of predefined processes for details

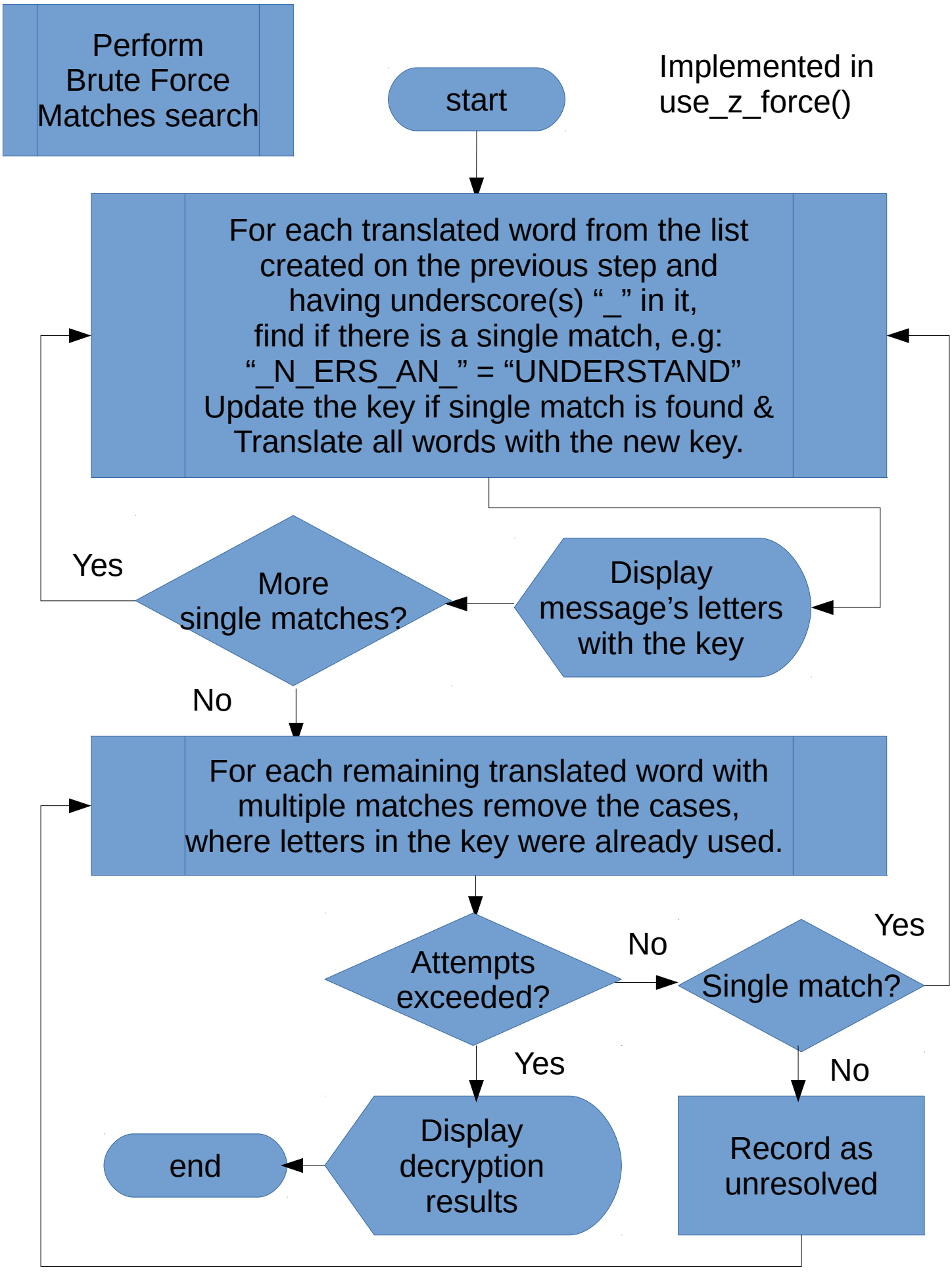



Access dictionary
start
Import sqlite3
If tables do not exist they will be created by Open function
Open en_dict.db
en_dict.db
No
Empty ?
Yes
Load en_dict.db
en_words.txt
Count records
Display count
end


Decipher message
start
Perform Finding 100% of Direct Hits (pages 4 &5)
Perform Brute Force Matches Search (page 6)
end


Perform Finding 100% of Direct Hits
start
Calculate patterns for all unique words in a message
Sort words & matching patterns in descending order by length
get_matches()
For each pattern find all matching words in a dictionary.
Identify the cases where for each word in matching pattern the letters are the same in the same column.
Record such cases as 100% matches hits and add them to the key.
A


Perform Finding 100% of Direct Hits
A
Will be used In the brute force attempt
Finalize the key which will include all 100% hits & "_" for missing letters
Translate original message using newly created key
Create new list containing pairs of words from the message and its translation
Will be used In the brute force attempt
Sort the list by words' length in descending order
Display string of unique letters & the matching key
end


Perform
Brute Force
Matches search
start
Implemented in
use_z_force()
For each translated word from the list
created on the previous step and
having underscore(s) "_" in it,
find if there is a single match, e.g:
"_N_ERS_AN_" = "UNDERSTAND"
Update the key if single match is found &
Translate all words with the new key.
Yes
More
single matches?
Display
message's letters
with the key
No
For each remaining translated word with
multiple matches remove the cases,
where letters in the key were already used.
Attempts
exceeded?
No
Single match?
Yes
Yes
No
end
Display
decryption
results
Record as
unresolved