# Practical Task 4
# Signatures & Vector Space Model

## Preliminary Notes:

All notes and requirements of practical tasks 2&3 still apply!

## Task 1 – Signatures

1. Implement the usage of signatures as another search mode for the Boolean retrieval! Choose a suitable hash function and use appropriate data structures for the signatures! For the creation of the signatures you may use the parameters $F = 64, D = 4$. You should determine an optimal value for $m$ yourself.
2. Make it also possible for the signature search to use the conjunction and disjunction of two terms!
3. Make it possible to use the signature search via the following command line parameter:

| parameter | explanation |
|---|---|
| `--search-mode "signatures"` | Only used with Boolean retrieval. When used, it causes the search to use signatures. |

   This parameter should work in combination with parameters from previous tasks. An exemplary program call could look like this:
   `$python my_ir_system.py --model "bool" --search-mode "signatures" --documents "original" -query "somesearchterm"`
4. You can assume that stemming is not used in combination with signatures. Therefore you do not have to handle program calls that contain: `--search-mode "signatures"` and `--stemming`.

## Task 2 – Vector Space Model

1. Implement the search using the vector space model and inverted lists!
2. Use *tf.idf* for the generation of term weights! The generation of the query vector should be done according to Salton/Buckley (1988).
3. Use the base algorithm with inverted lists as it was presented in the lecture!
4. Make it possible to use the VSM search via the following command line parameter:

| parameter | explanation |
|---|---|
| `--model "vector"` | Specifies that the Vector Space Model should be used for the search. |

This parameter should work in combination with parameters from previous tasks. An exemplary program call could look like this:
`$python my_ir_system.py --model "vector" --documents "original" -query "somesearchterm"`

5. The `--search-mode` parameter is only intended to work with Boolean retrieval.Therefore you do not have to handle program calls that contain: `--model "vector"` and `--search-mode`.

The format of the program output via stdout remains the same as with the last task sheet. Please do not add additional `print()` output and stick to the requirements!

**Make sure that your solution considers all requirements listed in this file and upload it on Moodle until the specified deadline!**