SortItout is an plication which sorts input file to output one.. SortItout (further I call it SIT) implements 2 main algorithms: Direct and Indirect.

- Direct algorithm allows to sort directly in a RAM and has about O(log(n)) complexity.
- Indirect algorithm allows to sort files much bigger then RAM and has about O(n*log(n)) complexity.

SIT should get 3 parameters: 1st is “input file name”, 2d is “output file name”, 3d is “size of RAM in Mb allowed to use”.

Only first parameter is critical, the rest are optional.

```
USAGE of the ./sortItout:
./sortItout --help or -h show this message
./sortItout <inputFile>
./sortItout <inputFile>  <outputFile>
./sortItout <inputFile>  <outputFile>  [memsize] Mb  sorts input file to output one
enter parameters please!
```

Input file must exist. If the rest of parameters didn't enter they took default values.

Memsize – is optional and has recommendation meaning, it may not bigger then free RAM, but it may be less.
By default SIT tries to use about 80% of free RAM.
In case of “indirect sort” the algorithm tries to split input file in to a number of files which less then Memsize. Then SIT merge them back in one sorted file.

Diagram direct algorithm:

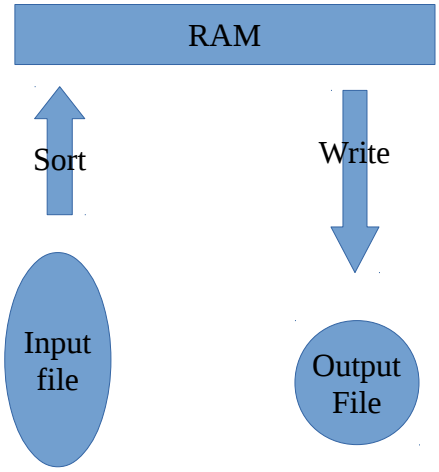
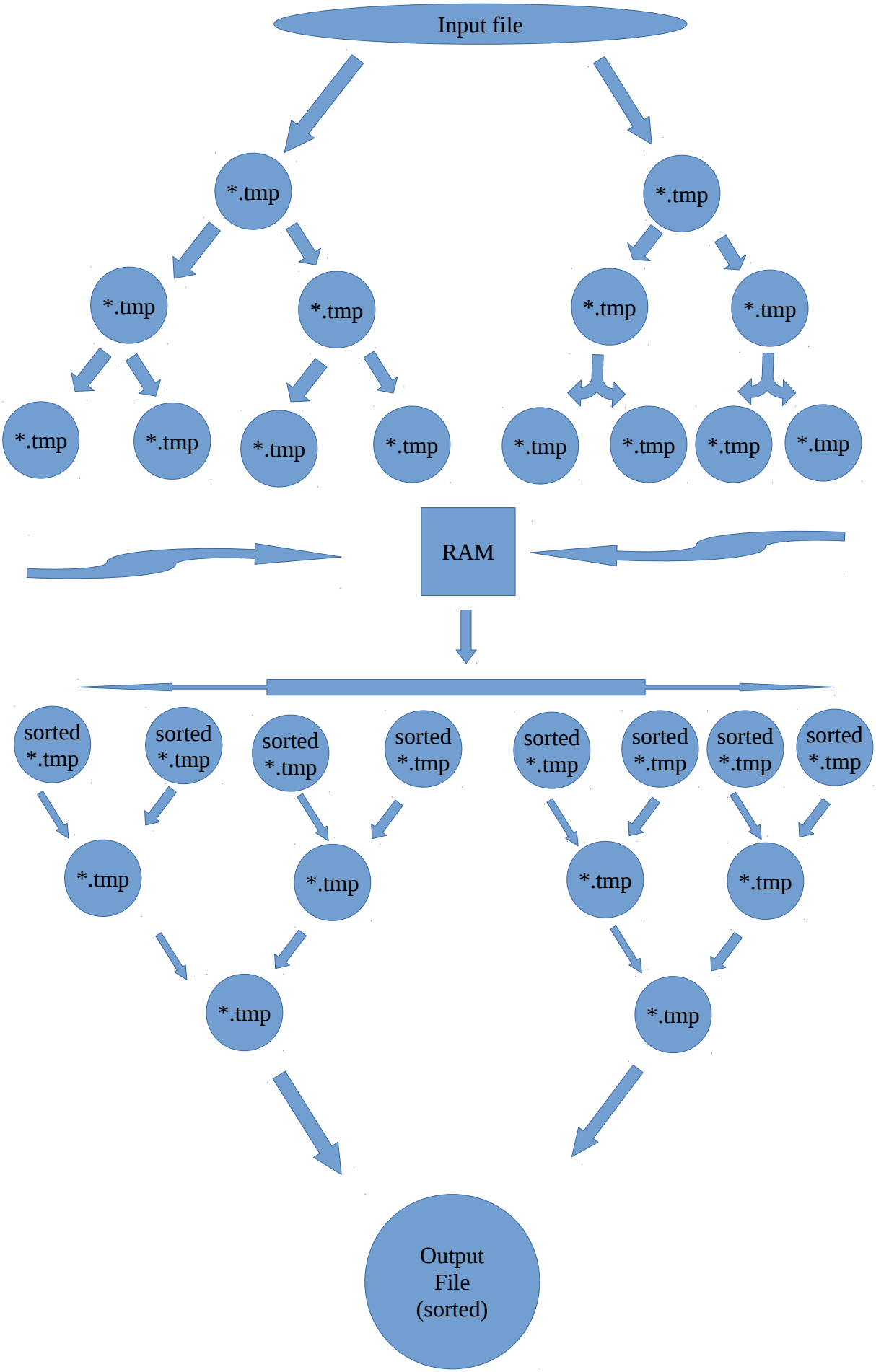


Diagram indirect algorithm:



NOTES:

File should consist from text lines divided by “\n”.
Text line mustn't have white spaces.
One line mustn't be bigger then RAM limitation.

SIT consist from: main.cpp fman.hpp fman.cpp cmdparcer.hpp cmdparcer.cpp makefile.

To build:
1 extract tar to your folder
2 cd to folder name
3 use “make” to build SIT

SIT was build with gcc version 4.8.4 (your gcc should support c++11)