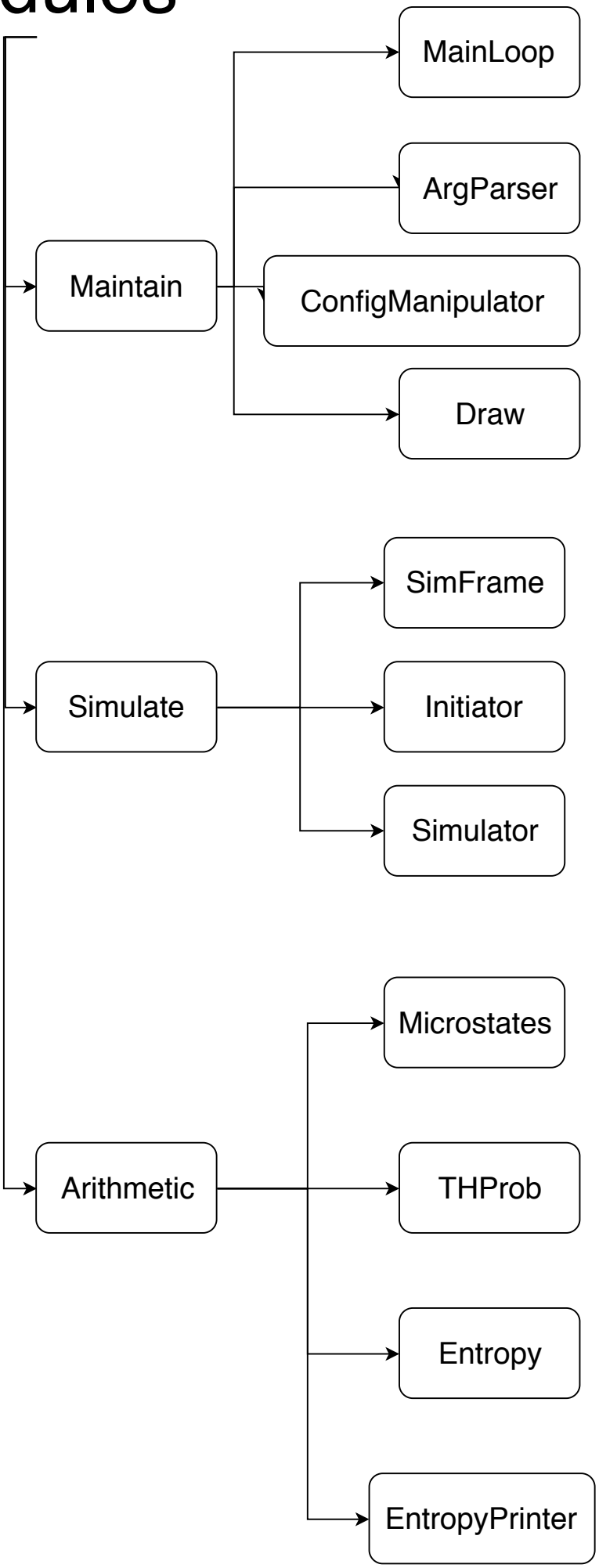
Modules


MainLoop
ArgParser
Maintain
ConfigManipulator
Draw
SimFrame
Simulate
Initiator
Simulator
Microstates
Arithmetic
THProb
Entropy
EntropyPrinter

Maintain

Part of program which embrace all modules into one working program

ArgParser

Is a part of program, taking part in analysing
user supplied startup arguments

Proper arguments are:
-f file -> read particles positions and velocities from file
-m time -> evaluate only microstates
-t time -> evaluate only thermal probability
-e time -> evaluate only entropy
-c file -> create chart with entropy
-r -> recreate config file
-s -> save all output to file
-T -> set simulation period

	constructor(argv:str[])	

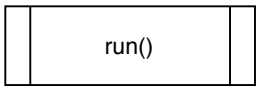
Takes: argv, which are arguments typed by user

	get_arguments()	

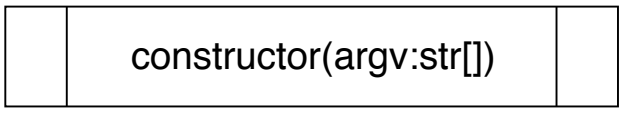
taeks: ALL arguments given by user
Returns: ArgumentParse.Namespace

MainLoop

Acts as main() in normal program, runs other function etc.



Takes: nth
Return: nth



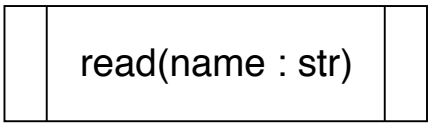
Takes: argv, which are arguments typed by user

ConfigManipulator

Is used as interface to communicate with config file, which store basic data with simulate configuration

Stored data are:

- sizeX : float -> contains width of simulated area
- sizeY : float -> contains height of simulated area
- maxSpeed : float -> define maximal speed of single particle
- particleAmount : int -> define amount of particles
- boxSize : float -> define size of one side of box
- time : int -> define time of simulation
- timeDelta -> smallest time simulation take part
- init_state_file : str -> file with initial positions and velocities for par



Takes: name of field described above
Returns: value of field

ticles

Simulate

Has code for conducting simulation of gas

SimFrme

Is class to store
one steep of simulation

get_particle(number:int)

Takes: number, unique number for each particle
Returns: Particle object, with corresponding number

get_particles()

Takes: nth
Returns: All possessed Particle objects as array

Particle

Single particle in simulation

get_position()

get_velocity()

get_box_position()

Initiator

Create initial frame of simulation,
exact demands of first frame are provided
in Schawiola materials

	constructor(file_name:str)	

Constructor should determine if file_name
if given and should read from file,
or if it's not it should randomize it

	create()	

Takes: nth
Returns: SimFrame

Simulator

Takes one frame,
simulate behavior of particles after elapsing some deltaTime,
return new SimFrame

	simulate(frame: SimFrame deltaTime : float)	

Takes: frame which is SimFrame type
time which is float type
Returns: frame which is SimFrame type

Arithmetic

Contains functions run on top of simulation to make some statistics about it

Microstates

Class describe microstate after elapsing some time.
Time should be taken from config file
Algorithm for evaluating microstate is carefully described in Schawiola materials

evaluate()

Takes: nth
Returns: array with ints describing microstates

THProb

Class describe thermodynamic probability after elapsing some time.
Time should be taken from config file
Algorithm for evaluating THP is carefully described in Schawiola materials

evaluate()

Takes: nth
Returns: simple int with THP

THProb

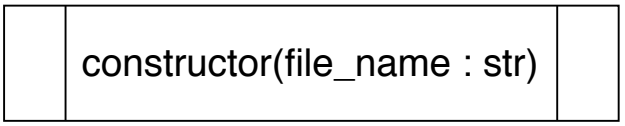
Class describe entropy after elapsing some time.
Time should be taken from config file
Algorithm for evaluating entropy is carefully described in Schawiola materials

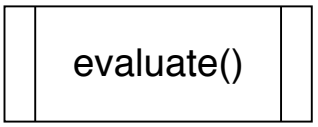
evaluate()

Takes: nth
Returns: simple int with entropy

EntropyPrinter

Class create chart describing
change of entropy in time
.Time should be taken from config file





Takes: nth
Returns: nth