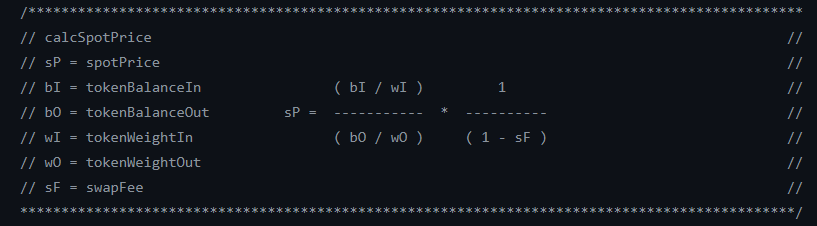
BPool –

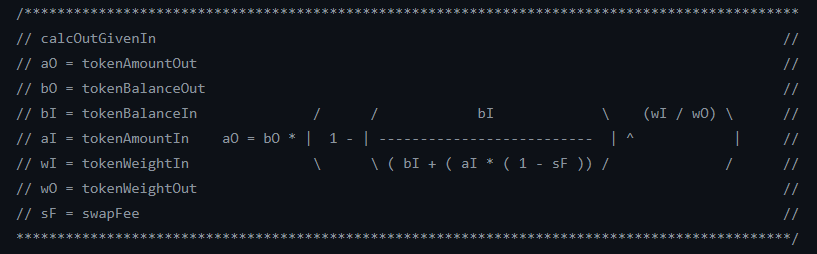
This contract uses BNUM, BMATH and all other balancer core contract directly or indirectly.

The main contract this Bpool uses is BMath which provide all the necessary calculations. Following types of calculations are there -

caclSpotPrice -



calcOutGivenIn -



Structs –

struct Record {

bool bound; // is token bound to pool

uint index; // private

uint denorm; // denormalized weight

uint balance;

}

Variables –

bool private \_mutex; - This is used in modifier for re-entrancy.

address private \_factory; // BFactory address to push token exitFee to

address private \_controller; // has CONTROL role , this contract is the which decides the pool parameters are fixed or not, means it public or private.

bool private \_publicSwap; // true if PUBLIC can call SWAP functions

uint private \_swapFee; - to store the swap fees.

bool private \_finalized; - to check if a pools parameters are finalized or not.

// `setSwapFee` and `finalize` require CONTROL

// `finalize` sets `PUBLIC can SWAP`, `PUBLIC can JOIN`

address[] private \_tokens;

mapping(address=>Record) private \_records;

uint private \_totalWeight;