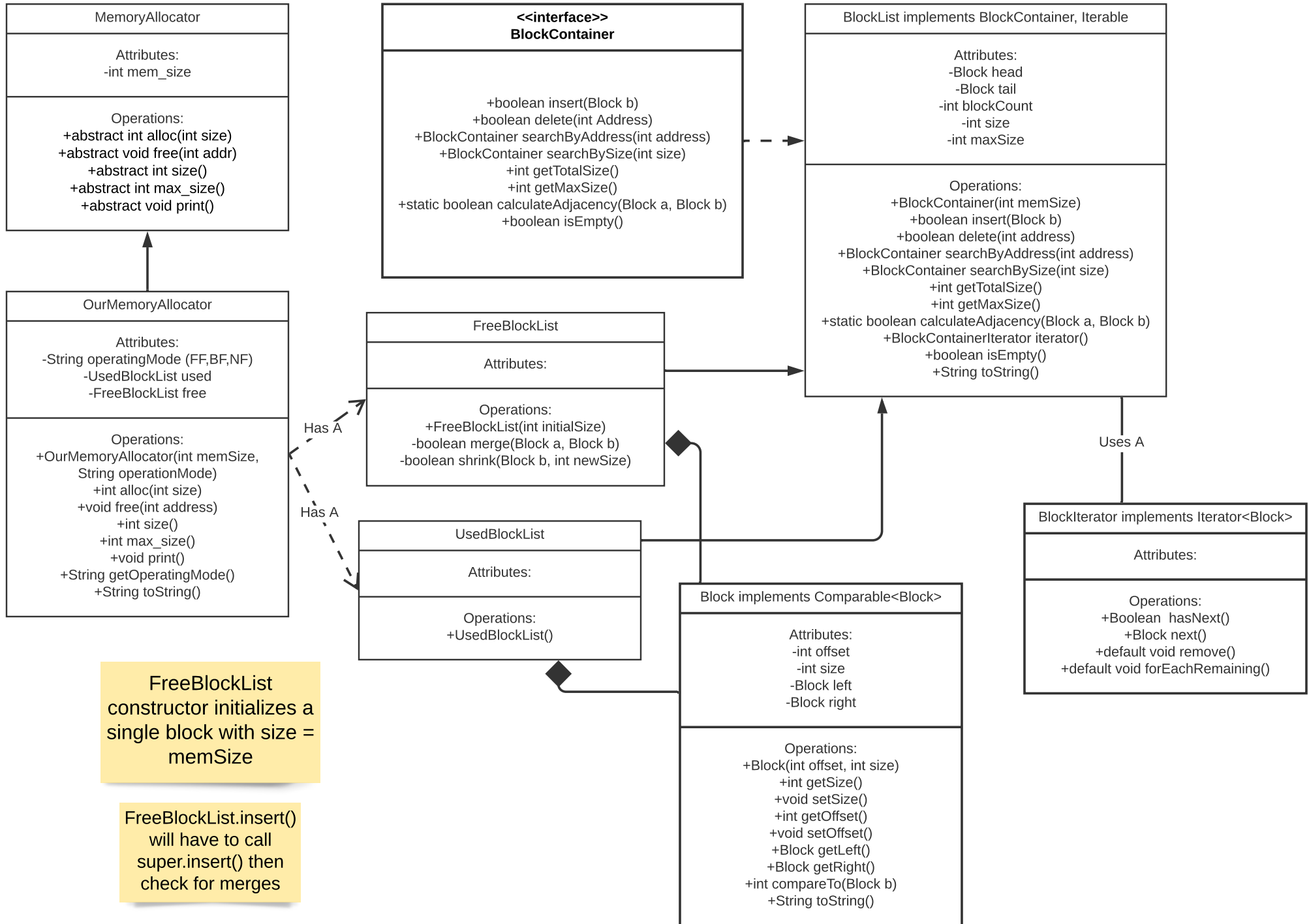**BlockContainer.calculateAdjacency() determines adjacency based on offset and size**

The search methods return BlockContainers with *shallow* copies of all blocks that match the search criteria, or empty BlockContainer if none

---

MemoryAllocator

Attributes:
-int mem_size

Operations:
+abstract int alloc(int size)
+abstract void free(int addr)
+abstract int size()
+abstract int max_size()
+abstract void print()

<<interface>>
BlockContainer

+boolean insert(Block b)
+boolean delete(int Address)
+BlockContainer searchByAddress(int address)
+BlockContainer searchBySize(int size)
+int getTotalSize()
+int getMaxSize()
+static boolean calculateAdjacency(Block a, Block b)
+boolean isEmpty()

BlockList implements BlockContainer, Iterable

Attributes:
-Block head
-Block tail
-int blockCount
-int size
-int maxSize

Operations:
+BlockContainer(int memSize)
+boolean insert(Block b)
+boolean delete(int address)
+BlockContainer searchByAddress(int address)
+BlockContainer searchBySize(int size)
+int getTotalSize()
+int getMaxSize()
+static boolean calculateAdjacency(Block a, Block b)
+BlockContainerIterator iterator()
+boolean isEmpty()
+String toString()

OurMemoryAllocator

Attributes:
-String operatingMode (FF,BF,NF)
-UsedBlockList used
-FreeBlockList free

Operations:
+OurMemoryAllocator(int memSize, String operationMode)
+int alloc(int size)
+void free(int address)
+int size()
+int max_size()
+void print()
+String getOperatingMode()
+String toString()

FreeBlockList

Attributes:

Operations:
+FreeBlockList(int initialSize)
-boolean merge(Block a, Block b)
-boolean shrink(Block b, int newSize)

Has A

Has A

UsedBlockList

Attributes:

Operations:
+UsedBlockList()

Block implements Comparable<Block>

Attributes:
-int offset
-int size
-Block left
-Block right

Operations:
+Block(int offset, int size)
+int getSize()
+void setSize()
+int getOffset()
+void setOffset()
+Block getLeft()
+Block getRight()
+int compareTo(Block b)
+String toString()

Uses A

BlockIterator implements Iterator<Block>

Attributes:

Operations:
+Boolean hasNext()
+Block next()
+default void remove()
+default void forEachRemaining()

FreeBlockList constructor initializes a single block with size = memSize

FreeBlockList.insert() will have to call super.insert() then check for merges