Can python access heap ?

In Python, you don't have direct access to the heap like you do in lower-level languages such as C or C++. Python manages memory automatically through its memory manager. Python uses a system of memory allocation called the Python Memory Manager, which handles the allocation and deallocation of memory for your Python programs.

While you can use libraries like 'ctypes' or 'cffi' to interface with C code that manages memory on the heap, directly accessing the Python heap is not recommended. Python's memory management is designed to be abstracted away from the programmer, and manual memory management can lead to memory leaks, corruption, or other serious issues.

What's multithreading?

Multithreading is a concurrent execution model where multiple threads run independently within the same process, sharing the same resources such as memory space but having their own separate execution paths. A thread is the smallest unit of execution that a program can schedule. Multithreading allows for concurrent execution of tasks and can improve the overall performance and responsiveness of a program.

In a single-threaded program, tasks are executed sequentially, one after the other. In a multithreaded program, multiple threads can run concurrently, performing different tasks simultaneously. Each thread has its own program counter, register set, and stack, but they share the same data space.

Thread: A thread is the smallest unit of execution within a process. It has its own program counter, registers, and stack.

Multithreading: The concurrent execution of multiple threads within the same process.