**Algorithm 1** Memory Management with CHERI

**Define** FILENAME $\leftarrow$ "$/dev/contigmem$"
**Define** $ptr$ ▷ Pointer to allocated memory
**Define** $MallocCounter$ ▷ Remaining memory size
**Define** $sizeUsed$ ▷ Size of used memory
**function** INITALLOC
    $sz \leftarrow 100000000$ ▷ Set allocation size
    $fd \leftarrow \text{open}(FILENAME, O\_RDWR, 0600)$
    **if** $fd < 0$ **then**
        **Handle Error:** exit(EXIT_FAILURE)
    **end if**
    $offset \leftarrow 0$ ▷ Offset for seeking
    **if** $\text{ftruncate}(fd, sz) < 0$ **then**
        **Handle Error:** close(fd), exit(EXIT_FAILURE)
    **end if**
    $ptr \leftarrow \text{mmap}(NULL, sz, PROT\_READ|PROT\_WRITE, MAP\_SHARED, fd, 0)$
    **if** $ptr = \text{MAP\_FAILED}$ **then**
        **Handle Error:** exit(EXIT_FAILURE)
    **end if**
    $MallocCounter \leftarrow sz$
**end function**
**function** MALLOC(sz)
    $sz \leftarrow \text{align\_up}(sz, \text{\_Alignof}(max\_align\_t))$ ▷ Align the size
    $MallocCounter \leftarrow MallocCounter - sz$
    $ptrLink \leftarrow ptr[MallocCounter]$ ▷ Allocate memory at the current position
    $ptrLink \leftarrow \text{cheri\_setbounds}(ptrLink, sz)$ ▷ Set bounds for CHERI
    **return** $ptrLink$ ▷ Return the allocated pointer
**end function**
**function** FREECHERI(ptr)
    $len \leftarrow \text{cheri\_getlen}(ptr)$ ▷ Get the length from pointer bounds
    munmap(ptr, len) ▷ Unmap the memory
**end function**