





Some pointer arithmetic and is also possible with buffers.

Figure 16. Algorithm for Removing a Buffer

When removing a buffer, the first step is to ensure that the buffer is not empty. The first step is to check the buffer's size. If the buffer is empty, the program can return. If the buffer is not empty, the program can proceed to remove the buffer. The program can use the buffer's size to determine the number of elements to remove. The program can then use a loop to remove the elements from the buffer. The program can then return the buffer's size to the caller.



Whiteboard content:

- Aspirin synthesis
- Reaction scheme with chemical structures
- Labels: "initial system call", "system call", "initial task", "return of buffer", "return"
- Flowchart on the left: "Aspirin" -> "read from" -> "D/D" -> "last activity"

Projection screen content:

Figure 14. Algorithm for creating a buffer

Before entering the main program of an operating system, the user must first enter the user's name and password. The kernel then reads the user's name and password from the file. The kernel then reads the buffer from the file and stores it in the buffer. When the kernel reads the buffer, it stores the buffer according to the algorithm shown in Figure 14. The kernel then reads the buffer from the file and stores it in the buffer. When the kernel reads the buffer, it stores the buffer according to the algorithm shown in Figure 14. The kernel then reads the buffer from the file and stores it in the buffer. When the kernel reads the buffer, it stores the buffer according to the algorithm shown in Figure 14.

