

F Prime Static Memory Allocation Flow

```
// Step 1: Request Memory & Check Pool
bool isPoolAvailable(size_t requested_size) const {
    return !is_allocated && (requested_size <=
    POOL_SIZE);
}
```

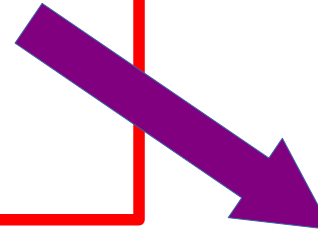


Request
Memory

Pool
Available?

```
// Step 2: Allocate Block
uint8_t* allocateBlock(size_t size) {
    if (!isPoolAvailable(size)) {
        return nullptr; // Error case - would trigger error handler
    }

    is_allocated = true;
    return pool; // Return pointer to allocated memory
}
```



Yes

No

Allocate
Block

Error
Handler

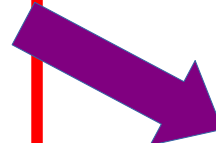
```
// Step 3: Return Buffer (implicitly part of allocateBlock)
```

```
// Step 4: Use Memory (handled by client code)
```

Return
Buffer

Client Code
Use Memory

```
// Step 5: Deallocate
void deallocate() {
    is_allocated = false;
}
```



Deallocate
Triggered by: exit,
client call, shutdown

