Memory Leaks and Dangling Pointers

Lecture 5
Secs 2.4, 3.4

Robb T. Koether

Hampden-Sydney College

Mon, Jan 26, 2009

Outline

Memory Leaks

Definition (Memory Leak)

A memory leak occurs when all pointers to a block of allocated memory have been lost.

- Leaked memory cannot be accessed or reallocated; it is useless.
- Excessive memory leaks may cause the program to run out of usable memory and crash.
- Memory leaks should *always* be avoided.

Dangling Pointers

Definition (Dangling Pointer)

A dangling pointer is a non-null pointer that points to unallocated memory.

- Dereferencing a dangling pointer may cause the program to crash.
- We do not necessarily avoid dangling pointers, but we must be careful.

Avoiding Dangling Pointers

- It impossible to test a non-null pointer to see whether it is dangling.
- Always set pointers to `NULL` if they do not point to allocated memory.
- Then compare them to `NULL` to see whether they point to allocated memory.

Example

Example (Avoiding Memory Leaks)

- The `setSize()` function of the `Vectr` class must
  - Allocate new memory of the specified size.
  - Copy the old values into the new memory.
  - Deallocate the memory that the `Vectr` is currently using.
  - Redirect the `Vectr` to the new memory.

Example

Example (Avoiding Memory Leaks)

- The `input()` function of the `Vectr` class must
  - Deallocate the memory that the `Vectr` is currently using (if any).
  - Allocate new memory for the values to be input.
  - Continue to increase the allocated memory as more values are read.

Example

Example (The `Vectr` Class)

- Download `vectr.h`.
- Download `vectr.cpp`.
- Download and run `VectrTest.cpp`.

Assignment

Homework

- Read Section 2.4, page 69.
- Read Section 3.4, pages 120 - 121.