

Lab 9: Traversing Directories

The exercises in this section develop programs to traverse directory trees in depth-first and breadth-first orders. Depth-first searches explore each branch of a tree to its leaves before looking at other branches. Breadth-first searches explore all the nodes at a given level before descending lower in the tree.

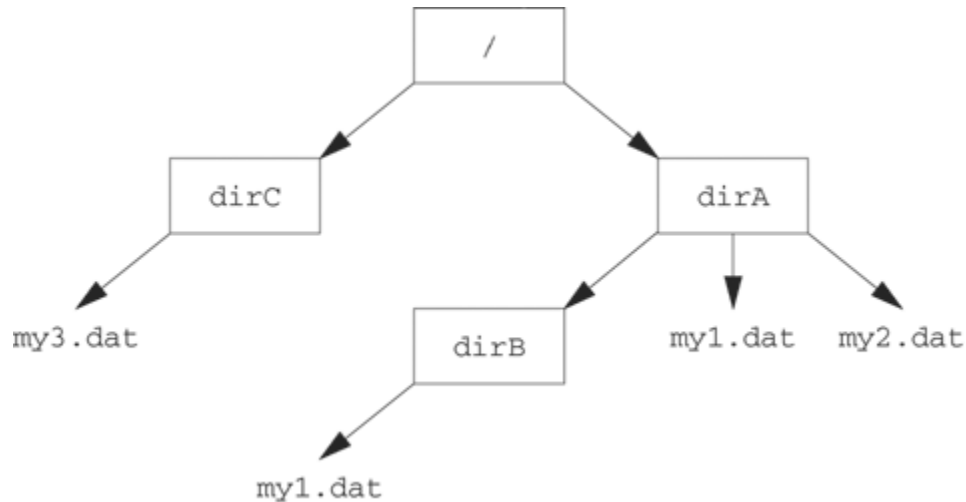


Figure 10.1. Tree structure of a file system

Task 1: Traverse directory tree in depth-first order.

For the file system tree in Figure 10.1, depth-first ordering visits the nodes in the following order.

```
/
  dirC
    my3.dat
  dirA
    dirB
      my1.dat
    my1.dat
    my2.dat
```

The indentation of the filenames in the above example shows the level in the file system tree. Depth-first search is naturally recursive, as indicated by the following pseudocode.

```
depthfirst(root) {
  for each node at or below root
    visit node;
    if node is a directory
      depthfirst(node);
}
```

Task 2: Traverse directory tree in breadth-first order.

For the file system tree in Figure 10.1, breadth-first order visits the nodes in the following order.

```
/
/dirC
/dirA
/dirC/my3.dat
/dirA/dirB
/dirA/my1.dat
/dirA/my2.dat
/dirA/dirB/my1.dat
```

Breadth-first search can be implemented with a queue similar to the history queue of [Program 2.8](#) on page 47 of the text book. As the program encounters each directory node at a particular level, it enqueues the complete pathname for later examination. The following pseudocode assumes the existence of a queue. The `enqueue` operation puts a node at the end of the queue, and the `dequeue` operation removes a node from the front of the queue.

```
breadthfirst(root) {
    enqueue(root);
    while (queue is not empty) {
        dequeue(&next);
        for each node directly below next:
            visit the node
            if node is a directory
                enqueue(node)
    }
}
```

Task 3: Implement the pfind utility

NAME

find - search for files in a directory hierarchy

SYNOPSIS

find [path...] [expression]

DESCRIPTION

This manual page documents the GNU version of find. find searches the directory tree rooted at each given file name by evaluating the given expression from left to right, according to the rules of precedence (see section OPERATORS), until the outcome is known (the left hand side is false for and operations, true for or), at which point find moves on to the next file name.

Create pfind utility to search the files