

In-Class Problems: Pathname Translation

Write a pseudo-code procedure to translate a pathname into an i-number in a hierarchical file system. Your procedure should take a pathname (of type string) as its only parameter. It should return the i-number (of type integer) of the file specified by the input pathname. If any error is encountered during translation, your function should return the special code *INVALID* instead of a real i-number. Assume that the i-number of the root directory is 0. To simplify the pseudo-code, treat strings like primitive data types.

Your procedure should use the following functions:

boolean is_directory(int i-number) This function returns the value “TRUE” if the file whose i-number is given is a directory. It returns “FALSE” otherwise.

int num_components(string pathname) This function returns the number of components in the specified pathname. For example, num_components(“/a/b/c”) returns 3, num_components(“/foo”) returns 1, and num_components(“/”) returns 0.

string get_component(int i, string pathname) This function returns a string representing the *i*th component of the specified pathname. The number “i” should be a positive integer. For example, get_component(2, “/a/b/c”) will return the string “b”, and get_component(1, “/foo/bar”) will return “foo”. If there is no *i*th component, the function returns the empty string “”.

int dsearch(int i-number, string component) This function is used to search a directory file for an element whose name matches the “component” argument. The parameter “i-number” must be the i-number of the directory file to be searched. The function returns the i-number of the matching entry. For example, the call dsearch(10, “foo”) will return the i-number of the “foo” entry in the directory file whose i-number is 10. If there is no such entry in the specified directory, the function returns the value -1.