**FsMkdir**

* [Example 1](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#mkdir-ex-1)

Program:

int main(void) {

Fs fs = FsNew();

*// this is equivalent to FsMkdir(fs, "tmp"); because initially,*

*// the current working directory is the root directory*

FsMkdir(fs, "/tmp");

FsMkdir(fs, "/tmp/tmp.123");

FsMkdir(fs, "/usr");

FsMkdir(fs, "/bin");

*// see the section for FsTree for details*

FsTree(fs, NULL);

}

Expected output:

/

bin

tmp

tmp.123

usr

* [Example 2](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#mkdir-ex-2)

Program:

int main(void) {

Fs fs = FsNew();

FsMkdir(fs, "/tmp");

FsMkdir(fs, "tmp");

FsMkdir(fs, "./tmp");

}

Expected output:

mkdir: cannot create directory 'tmp': File exists

mkdir: cannot create directory './tmp': File exists

* [Example 3](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#mkdir-ex-3)

Program:

int main(void) {

Fs fs = FsNew();

*// see the section for FsMkfile for details*

FsMkfile(fs, "hello.txt");

FsTree(fs, NULL);

FsMkdir(fs, "hello.txt/world");

FsMkdir(fs, "html");

FsMkfile(fs, "html/index.html");

FsMkdir(fs, "html/index.html/hi");

FsTree(fs, NULL);

}

Expected output:

/

hello.txt

mkdir: cannot create directory 'hello.txt/world': Not a directory

mkdir: cannot create directory 'html/index.html/hi': Not a directory

/

hello.txt

html

index.html

* [Example 4](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#mkdir-ex-4)

Program:

int main(void) {

Fs fs = FsNew();

FsMkdir(fs, "hello/world");

}

Expected output:

mkdir: cannot create directory 'hello/world': No such file or directory

* [Example 5](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#mkdir-ex-5)

Program:

int main(void) {

Fs fs = FsNew();

FsMkdir(fs, ".");

FsMkdir(fs, "..");

}

Expected output:

mkdir: cannot create directory '.': File exists

mkdir: cannot create directory '..': File exists

**FsMkfile**

[Example 1](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#mkfile-ex-1)

Program:

int main(void) {

Fs fs = FsNew();

*// this is equivalent to FsMkdir(fs, "hello.c"); because initially,*

*// the current working directory is the root directory*

FsMkfile(fs, "/hello.c");

FsMkfile(fs, "world.c");

FsMkdir(fs, "/bin");

FsMkfile(fs, "bin/mkdir");

FsMkfile(fs, "bin/mkfile");

*// see the section for FsTree for details*

FsTree(fs, NULL);

}

Expected output:

/

bin

mkdir

mkfile

hello.c

world.c

* [Example 2](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#mkfile-ex-2)
* Program:
* int main(void) {
* Fs fs = FsNew();
* FsMkfile(fs, "/tmp");
* FsMkfile(fs, "tmp");
* FsMkfile(fs, "./tmp");
* }
* Expected output:
* mkfile: cannot create file 'tmp': File exists
* mkfile: cannot create file './tmp': File exists
* [Example 3](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#mkfile-ex-3)

Program:

int main(void) {

Fs fs = FsNew();

FsMkfile(fs, "hello");

FsTree(fs, NULL);

FsMkfile(fs, "hello/world");

FsMkdir(fs, "html");

FsMkfile(fs, "html/index.html");

FsMkfile(fs, "html/index.html/hi");

FsTree(fs, NULL);

}

Expected output:

/

hello

mkfile: cannot create file 'hello/world': Not a directory

mkfile: cannot create file 'html/index.html/hi': Not a directory

/

hello

html

index.html

* [Example 4](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#mkfile-ex-4)
* Program:
* int main(void) {
* Fs fs = FsNew();
* FsMkfile(fs, "hello/world");
* }
* Expected output:
* mkfile: cannot create file 'hello/world': No such file or directory
* [Example 5](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#mkfile-ex-5)
* Program:
* int main(void) {
* Fs fs = FsNew();
* FsMkfile(fs, ".");
* FsMkfile(fs, "..");
* }
* Expected output:
* mkfile: cannot create file '.': File exists
* mkfile: cannot create file '..': File exists

#### FsCd

* [Example 1](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#cd-ex-1)

Program:

int main(void) {

Fs fs = FsNew();

FsMkdir(fs, "/home");

FsCd(fs, "home");

FsMkdir(fs, "jas");

FsCd(fs, "jas");

FsMkdir(fs, "cs2521");

FsCd(fs, "cs2521");

FsMkdir(fs, "lectures");

FsMkdir(fs, "tutes");

FsMkdir(fs, "labs");

FsTree(fs, NULL);

}

Expected output:

/

home

jas

cs2521

labs

lectures

tutes

* [Example 2](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#cd-ex-2)
* Program:
* int main(void) {
* Fs fs = FsNew();
* FsCd(fs, "."); *// does nothing*
* FsCd(fs, ".."); *// does nothing, since the parent of the root directory is itself*
* FsCd(fs, "./.././.././"); *// also does nothing*
* FsMkdir(fs, "tmp");
* FsCd(fs, "tmp");
* FsMkfile(fs, "random.txt");
* FsMkdir(fs, "../bin");
* FsMkdir(fs, "./../home");
* FsTree(fs, NULL);
* }
* Expected output:
* /
* bin
* home
* tmp
* random.txt
* [Example 3](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#cd-ex-3)

Program:

int main(void) {

Fs fs = FsNew();

FsMkdir(fs, "tmp");

FsCd(fs, "tmp");

FsCd(fs, NULL);

FsMkfile(fs, "hello.txt");

FsTree(fs, NULL);

}

Expected output:

/

hello.txt

tmp

* [Example 4](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#cd-ex-4)

Program:

int main(void) {

Fs fs = FsNew();

FsMkfile(fs, "hello");

FsCd(fs, "hello");

FsCd(fs, "hello/world");

}

Expected output:

cd: 'hello': Not a directory

cd: 'hello/world': Not a directory

* [Example 5](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#cd-ex-5)

Program:

int main(void) {

Fs fs = FsNew();

FsMkdir(fs, "tmp");

FsCd(fs, "bin");

FsCd(fs, "tmp/dir123");

}

Expected output:

cd: 'bin': No such file or directory

cd: 'tmp/dir123': No such file or directory

#### FsLs

* [Example 1](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#ls-ex-1)
* Program:
* int main() {
* Fs fs = FsNew();
* printf("---**\n**"); *// marker to separate output*
* FsLs(fs, "/");
* printf("---**\n**");
* FsMkfile(fs, "hello.txt");
* FsMkdir(fs, "tmp");
* FsLs(fs, "/");
* }
* Expected output:
* ---
* ---
* hello.txt
* tmp
* [Example 2](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#ls-ex-2)

Program:

int main(void) {

Fs fs = FsNew();

FsMkfile(fs, "hello.txt");

FsMkdir(fs, "tmp");

FsMkfile(fs, "tmp/world.txt");

FsLs(fs, "hello.txt");

FsLs(fs, "tmp/world.txt");

FsLs(fs, "tmp/.././hello.txt");

}

Expected output:

hello.txt

tmp/world.txt

tmp/.././hello.txt

* [Example 3](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#ls-ex-3)
* Program:
* int main(void) {
* Fs fs = FsNew();
* FsMkdir(fs, "tmp");
* FsMkfile(fs, "tmp/hello.txt");
* FsMkfile(fs, "tmp/world.txt");
* FsCd(fs, "tmp");
* FsLs(fs, NULL);
* }
* Expected output:
* hello.txt
* world.txt
* [Example 4](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#ls-ex-4)

Program:

int main(void) {

Fs fs = FsNew();

FsMkfile(fs, "hello");

FsLs(fs, "hello/world");

FsLs(fs, "hello/.");

}

Expected output:

ls: cannot access 'hello/world': Not a directory

ls: cannot access 'hello/.': Not a directory

* [Example 5](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#ls-ex-5)
* Program:
* int main(void) {
* Fs fs = FsNew();
* FsMkdir(fs, "tmp");
* FsLs(fs, "hello");
* FsLs(fs, "tmp/world");
* }
* Expected output:
* ls: cannot access 'hello': No such file or directory
* ls: cannot access 'tmp/world': No such file or directory

#### FsPwd

* [Example 1](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#pwd-ex-1)
* Program:
* int main() {
* Fs fs = FsNew();
* FsPwd(fs);
* FsMkdir(fs, "home");
* FsCd(fs, "home");
* FsPwd(fs);
* FsMkdir(fs, "jas");
* FsCd(fs, "jas");
* FsPwd(fs);
* }
* Expected output:
* /
* /home
* /home/jas
* [Example 2](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#pwd-ex-2)
* Program:
* int main() {
* Fs fs = FsNew();
* FsPwd(fs);
* FsMkdir(fs, "home");
* FsCd(fs, "home");
* FsPwd(fs);
* FsMkdir(fs, "jas");
* FsCd(fs, "jas");
* FsPwd(fs);
* }
* Expected output:
* /
* /home
* /home/jas

#### FsTree

* [Example 1](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#tree-ex-1)
* Program:
* int main() {
* Fs fs = FsNew();
* FsMkfile(fs, "hello.txt");
* FsMkfile(fs, "world.txt");
* FsMkdir(fs, "bin");
* FsMkfile(fs, "bin/ls");
* FsMkfile(fs, "bin/pwd");
* FsMkdir(fs, "home");
* FsMkdir(fs, "home/jas");
* FsMkfile(fs, "home/jas/todo.txt");
* FsMkfile(fs, "home/jas/mail.txt");
* FsTree(fs, "/home/jas");
* printf("---**\n**"); *// marker to separate output*
* FsTree(fs, NULL);
* }
* Expected output:
* /home/jas
* mail.txt
* todo.txt
* ---
* /
* bin
* ls
* pwd
* hello.txt
* home
* jas
* mail.txt
* todo.txt
* world.txt
* [Example 2](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#tree-ex-2)
* Program:
* int main(void) {
* Fs fs = FsNew();
* FsMkfile(fs, "hello");
* FsTree(fs, "hello");
* FsTree(fs, "./hello/world");
* }
* Expected output:
* tree: 'hello': Not a directory
* tree: './hello/world': Not a directory
* [Example 3](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#tree-ex-3)
* Program:
* int main(void) {
* Fs fs = FsNew();
* FsMkdir(fs, "tmp");
* FsTree(fs, "hello");
* FsTree(fs, "tmp/world");
* }
* Expected output:
* tree: 'hello': No such file or directory
* tree: 'tmp/world': No such file or directory

#### FsPut

* [Example 1](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#put-ex-1)
* Program:
* int main() {
* Fs fs = FsNew();
* FsMkfile(fs, "hello.txt");
* FsPut(fs, "hello.txt", "hello**\n**");
* FsPut(fs, "./hello.txt", "world**\n**"); *// overwrites existing content*
* }
* This program has no expected output.
* [Example 2](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#put-ex-2)
* Program:
* int main(void) {
* Fs fs = FsNew();
* FsMkdir(fs, "hello");
* FsPut(fs, "hello", "random-message**\n**");
* FsPut(fs, ".", "random-message**\n**");
* FsPut(fs, "/", "random-message**\n**");
* }
* Expected output:
* put: 'hello': Is a directory
* put: '.': Is a directory
* put: '/': Is a directory
* [Example 3](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#put-ex-3)
* Program:
* int main(void) {
* Fs fs = FsNew();
* FsMkfile(fs, "hello");
* FsPut(fs, "hello/world", "random-message**\n**");
* }
* Expected output:
* put: 'hello/world': Not a directory
* [Example 4](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#put-ex-4)
* Program:
* int main(void) {
* Fs fs = FsNew();
* FsPut(fs, "hello/world", "random-message**\n**");
* }
* Expected output:
* put: 'hello/world': No such file or directory

#### FsCat

* [Example 1](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#cat-ex-1)
* Program:
* int main() {
* Fs fs = FsNew();
* FsMkfile(fs, "hello.txt");
* FsPut(fs, "hello.txt", "hello**\n**");
* FsCat(fs, "hello.txt");
* FsPut(fs, "./hello.txt", "world**\n**"); *// overwrites existing content*
* FsCat(fs, "/hello.txt");
* }
* Expected output:
* hello
* world
* [Example 2](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#cat-ex-2)
* Program:
* int main(void) {
* Fs fs = FsNew();
* FsMkdir(fs, "hello");
* FsCat(fs, "hello");
* FsCat(fs, ".");
* FsCat(fs, "/");
* }
* Expected output:
* cat: 'hello': Is a directory
* cat: '.': Is a directory
* cat: '/': Is a directory
* [Example 3](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#cat-ex-3)
* Program:
* int main(void) {
* Fs fs = FsNew();
* FsMkfile(fs, "hello");
* FsCat(fs, "hello/world");
* }
* Expected output:
* cat: 'hello/world': Not a directory
* [Example 4](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#cat-ex-4)
* Program:
* int main(void) {
* Fs fs = FsNew();
* FsCat(fs, "hello/world");
* }
* Expected output:
* cat: 'hello/world': No such file or directory

#### FsDldir

* [Example 1](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#dldir-ex-1)
* Program:
* int main() {
* Fs fs = FsNew();
* FsMkdir(fs, "hello");
* FsMkdir(fs, "hello/world");
* FsTree(fs, NULL);
* printf("---**\n**"); *// marker to separate output*
* FsDldir(fs, "hello/world");
* FsDldir(fs, "hello");
* FsTree(fs, NULL);
* }
* Expected output:
* /
* hello
* world
* ---
* /
* [Example 2](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#dldir-ex-2)
* Program:
* int main(void) {
* Fs fs = FsNew();
* FsMkdir(fs, "hello");
* FsMkdir(fs, "hello/world");
* FsDldir(fs, "hello");
* FsTree(fs, NULL);
* }
* Expected output:
* dldir: failed to remove 'hello': Directory not empty
* /
* hello
* world
* [Example 3](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#dldir-ex-3)
* Program:
* int main(void) {
* Fs fs = FsNew();
* FsMkfile(fs, "hello");
* FsDldir(fs, "hello/world");
* }
* Expected output:
* dldir: failed to remove 'hello/world': Not a directory
* [Example 4](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#dldir-ex-4)
* Program:
* int main(void) {
* Fs fs = FsNew();
* FsMkfile(fs, "hello");
* FsDldir(fs, "hello/world");
* }
* Expected output:
* dldir: failed to remove 'hello/world': Not a directory

#### FsDl

* [Example 1](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#dl-ex-1)
* Program:
* int main() {
* Fs fs = FsNew();
* FsMkdir(fs, "hello");
* FsMkfile(fs, "hello/world.txt");
* FsMkfile(fs, "abc.txt");
* FsTree(fs, NULL);
* printf("---**\n**"); *// marker to separate output*
* FsDl(fs, true, "abc.txt");
* FsTree(fs, NULL);
* printf("---**\n**"); *// marker to separate output*
* FsDl(fs, true, "hello");
* FsTree(fs, NULL);
* }
* Expected output:
* /
* abc.txt
* hello
* world.txt
* ---
* /
* hello
* world.txt
* ---
* /
* [Example 2](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#dl-ex-2)
* Program:
* int main(void) {
* Fs fs = FsNew();
* FsMkdir(fs, "hello");
* FsDl(fs, false, "hello");
* }
* Expected output:
* dl: cannot remove 'hello': Is a directory
* [Example 3](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#dl-ex-3)
* Program:
* int main(void) {
* Fs fs = FsNew();
* FsMkfile(fs, "hello");
* FsDl(fs, false, "hello/world");
* }
* Expected output:
* dl: cannot remove 'hello/world': Not a directory
* [Example 4](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#dl-ex-4)
* Program:
* int main(void) {
* Fs fs = FsNew();
* FsDl(fs, "hello/world");
* }
* Expected output:
* dl: cannot remove 'hello/world': No such file or directory

#### FsCp

* [Example 1](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#cp-ex-1)
* Program:
* int main() {
* Fs fs = FsNew();
* FsMkfile(fs, "hello.txt");
* FsPut(fs, "hello.txt", "hello**\n**");
* FsMkfile(fs, "world.txt");
* FsPut(fs, "world.txt", "world**\n**");
* FsCat(fs, "world.txt");
* printf("---**\n**");
* char \*src[] = { "hello.txt", NULL };
* FsCp(fs, false, src, "world.txt");
* FsCat(fs, "world.txt");
* printf("---**\n**");
* FsTree(fs, NULL);
* }
* Expected output:
* world
* ---
* hello
* ---
* /
* hello.txt
* world.txt
* [Example 2](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#cp-ex-2)
* Program:
* int main() {
* Fs fs = FsNew();
* FsMkfile(fs, "hello.txt");
* FsPut(fs, "hello.txt", "hello**\n**");
* FsMkfile(fs, "world.txt");
* FsPut(fs, "world.txt", "world**\n**");
* FsCat(fs, "world.txt");
* printf("---**\n**");
* char \*src[] = { "hello.txt", NULL };
* FsCp(fs, false, src, "world.txt");
* FsCat(fs, "world.txt");
* printf("---**\n**");
* FsTree(fs, NULL);
* }
* Expected output:
* world
* ---
* hello
* ---
* /
* hello.txt
* world.txt
* [Example 3](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#cp-ex-3)
* Program:
* int main() {
* Fs fs = FsNew();
* FsMkfile(fs, "hello.txt");
* FsPut(fs, "hello.txt", "hello**\n**");
* FsMkdir(fs, "world");
* FsMkfile(fs, "world/hello.txt");
* FsTree(fs, NULL);
* printf("---**\n**");
* FsCat(fs, "world/hello.txt");
* printf("---**\n**");
* char \*src[] = { "hello.txt", NULL };
* FsCp(fs, false, src, "world");
* FsTree(fs, NULL);
* printf("---**\n**");
* FsCat(fs, "world/hello.txt");
* }
* Expected output:
* /
* hello.txt
* world
* hello.txt
* ---
* ---
* /
* hello.txt
* world
* hello.txt
* ---
* hello
* [Example 4](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#cp-ex-4)
* Program:
* int main() {
* Fs fs = FsNew();
* FsMkdir(fs, "hello");
* FsMkfile(fs, "hello/a.txt");
* FsTree(fs, NULL);
* printf("---**\n**");
* char \*src[] = { "hello", NULL };
* FsCp(fs, true, src, "world");
* FsTree(fs, NULL);
* }
* Expected output:
* /
* hello
* a.txt
* ---
* /
* hello
* a.txt
* world
* a.txt
* [Example 5](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#cp-ex-5)
* Program:
* int main() {
* Fs fs = FsNew();
* FsMkdir(fs, "hello");
* FsMkfile(fs, "hello/a.txt");
* FsMkdir(fs, "world");
* FsTree(fs, NULL);
* printf("---**\n**");
* char \*src[] = { "hello", NULL };
* FsCp(fs, true, src, "world");
* FsTree(fs, NULL);
* }
* Expected output:
* /
* hello
* a.txt
* world
* ---
* /
* hello
* a.txt
* world
* hello
* a.txt

#### FsMv

* [Example 1](https://cgi.cse.unsw.edu.au/~cs2521/21T3/assignments/ass1#cp-ex-1)
* Program:
* int main() {
* Fs fs = FsNew();
* FsMkfile(fs, "hello.txt");
* FsPut(fs, "hello.txt", "hello**\n**");
* FsTree(fs, NULL);
* printf("---**\n**");
* char \*src[] = { "hello.txt", NULL };
* FsMv(fs, src, "world.txt");
* FsTree(fs, NULL);
* printf("---**\n**");
* FsCat(fs, "world.txt");
* }
* Expected output:
* /
* hello.txt
* ---
* /
* world.txt
* ---
* hello
* ---