

```

#include <iostream>
#include <cstdlib>
#include <cerrno>
#include <unistd.h>
#include <fcntl.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <time.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <sstream>
#include <pwd.h>
#include <grp.h>
#include <dirent.h>

```

```

using namespace std;

```

```

bool f = false;
bool r = false;

```

```

/**
 *Function used to remove directories
 *@param filename name of the directory to be recursively removed
 */
void rm(char * filename)
{
    struct stat sb;
    DIR* dir = opendir(filename);
    if(dir)// is it a directory
    {
        chdir(filename);
        struct dirent* dirent;
        while((dirent = readdir(dir)) != NULL)
        {
            char * argument = dirent->d_name;
            if((strcmp(argument, "..") ==0) || (strcmp(argument, ".")==0)) //dont
want to delete everything
            {
                continue;
            }
            else
            {
                if(stat(argument, &sb) != -1)
                {
                    if (S_ISDIR(sb.st_mode)) //recursive for directories
                    {
                        rm(argument);
                    }
                    else //if its a file
                    {

```

```

        if(f || (sb.st_mode & S_IWUSR) != 0)
        {
            unlink(argument);
        }
    else
    {
        cout << "rm: remove write-protected file '" << argument <<"'
? (Y/N):";

        string input;
        cin >> input;
        cout << "\n";
        if(input[0] == 'Y' || input[0] == 'y')
        {
            unlink(argument);
        }
        else
        {
            continue;
        }
    }

    } //if its a file
} //if stat != -1
else
{
    cout <<"Error: Directory '" << argument <<"' cannot be read"
<<endl;
}
} //end of else for rest of arguments, not . and ..
} //end of while
chdir("..");
if(f || (sb.st_mode & S_IWUSR) != 0)
{
    rmdir(filename);
}
else
{
    cout << "rm: remove write-protected directory '" << filename <<"' ?
(Y/N):";

    string input;
    cin >> input;
    cout << "\n";
    if(input[0] == 'Y' || input[0] == 'y')
    {
        rmdir(filename);
    }
}

} //if(dir)
else
{
    cout << "Error: cannot open directory '" << filename << "'" <<endl;
}
} //end of function

```

```

/**
 *Main method where everything is done
 *@param argc number of arguments
 *@param argv array of arguments
 */
int main(int argc, char * argv[])
{
    cout.setf(ios::unitbuf);
    struct stat sb;
    char * arguments[25];
    int argSize = 0;

    if(argc == 1)
    {
        cout << "Error: Enter a file to be removed" <<endl;
        return(EXIT_FAILURE);
    }

    for(int i = 0; i < argc; i++)
    {
        if (argv[i][0] == '-') //seeing if each arg starts with '-'
        {
            char * temp = argv[i];
            bool valid = false;
            for(uint j = 0; j < sizeof(temp); j++)//checking every spot of the arg
            that starts with '-'
            {
                if(temp[j] == 'r')
                {
                    r = true;
                    valid = true;
                }
                else if(temp[j] == 'f')
                {
                    f = true;
                    valid = true;
                }
            }
            if(valid == false)
            {
                cout << "Error: no -r or -f in option" << endl;
            }
        }
        else
        {
            arguments[argSize] = argv[i];
            argSize ++;
        }
    } //end of for loop

```

```

for(int i = 1; i < argSize; i++)
{
    char * argument = arguments[i];
    if(stat(argument, &sb) != -1)
    {
        if (S_ISDIR(sb.st_mode)) //argument is a directory
        {
            if(r)
            {
                if(f || (sb.st_mode & S_IWUSR) != 0)
                {
                    rm(argument);
                }
                else
                {
                    cout << "rm: remove write-protected directory '" << argument
<<"' ? (Y/N):";
                    string input;
                    cin >> input;
                    cout << "\n";
                    if(input[0] == 'Y' || input[0] == 'y')
                    {
                        rm(argument);
                    }
                    else
                    {
                        continue;
                    }
                }
            }
            else //directory is not recursive
            {
                cout << "rm: cannot remove '" << argument << "' : Is a directory"
<< endl;
            }
        }
        else //argument is a file
        {
            if(f || (sb.st_mode & S_IWUSR) != 0)
            {
                unlink(argument);
            }
            else
            {
                cout << "rm: remove write-protected file '" << argument <<"' ?
(Y/N):";
                string input;
                cin >> input;
                cout << "\n";
                if(input[0] == 'Y' || input[0] == 'y')
                {
                    unlink(argument);
                }
            }
        }
    }
}

```

```
        }
        else
        {
            continue;
        }
    }
}
else //if stat() == -1
{
    cout <<"Error: '" << argument << "'" could not be read!" << endl;
}
} //for loop
return EXIT_SUCCESS;
}
```