|  |
| --- |
| #include "types.h" |
|  | #include "stat.h" |
|  | #include "user.h" |
|  | #include "fcntl.h" |
|  | #include "fs.h" |
|  |  |
|  | int |
|  | delete(char \*path) |
|  | { |
|  | return unlink(path); |
|  | } |
|  |  |
|  | char\* |
|  | strcat(char \*d,char \*s) |
|  | { |
|  | char \*temp=d; |
|  | while(\*d) ++d; |
|  | while(\*s) \*d++=\*s++; |
|  | \*d=0; |
|  | return temp; |
|  | } |
|  |  |
|  | void |
|  | rm\_rf(char \*path) |
|  | { |
|  | char \*buff; |
|  | buff=(char\*)malloc(512\*sizeof(char)); |
|  | int fd0; |
|  | struct dirent de; |
|  | struct stat st; |
|  | if(path[strlen(path)-1]=='/') path[strlen(path)-1]=0; |
|  | if((fd0=open(path,0))<0) |
|  | { |
|  | printf(2,"rm: cannot open %s No such file or directory\n",path); |
|  | exit(); |
|  | } |
|  | if(fstat(fd0,&st)<0) |
|  | { |
|  | printf(2,"rm: cannot stat %s No such file or directory\n",path); |
|  | close(fd0); |
|  | exit(); |
|  | } |
|  | switch(st.type) |
|  | { |
|  | case T\_FILE: |
|  | { |
|  | delete(path); |
|  | break; |
|  | } |
|  | case T\_DIR: |
|  | { |
|  | strcpy(buff,path); |
|  | strcat(buff,"/"); |
|  | int len=strlen(buff); |
|  | while(read(fd0,&de,sizeof(de))==sizeof(de)) |
|  | { |
|  | if(de.inum==0 || de.name[0]=='.') continue; |
|  | memmove(buff+len,de.name,strlen(de.name)); |
|  | rm\_rf(buff); |
|  | memset(buff+len,'\0',sizeof(buff)+len); |
|  | } |
|  | unlink(path); |
|  | break; |
|  | } |
|  | } |
|  | free(buff); |
|  | close(fd0); |
|  | } |
|  |  |
|  | int main(int argc,char \*argv[]) |
|  | { |
|  | int i; |
|  | if(argc<2) |
|  | { |
|  | printf(2,"Usage: rm [OPTIONS] [files]...\n"); |
|  | printf(2,"Options:\n"); |
|  | printf(2," -rf : delete file secara recursive\n"); |
|  | exit(); |
|  | } |
|  | if(strcmp(argv[1],"-rf")==0) |
|  | { |
|  | for(i=2;i<argc;i++) |
|  | { |
|  | rm\_rf(argv[i]); |
|  | delete(argv[i]); |
|  | } |
|  | } |
|  | else |
|  | { |
|  | for(i=1;i<argc;i++) |
|  | { |
|  | delete(argv[i]); |
|  | } |
|  | } |
|  | exit(); |
|  | } |