How to solve the problem

Include library

```
#include <stdio.h>
#include <stdlib.h>
#include <stdbool.h>
#include <string.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <dirent.h>
#include <pwd.h>
#include <grp.h>
#include <errno.h>
#include <ctype.h>
```

Fetch required Info

- I used two struct to store the fetched information related to pid

```
typedef struct pidInfo
{
    int FD;
    char filename[256];
    int inode;
    struct pidInfo* next;
}FD;
typedef struct pidInfo_link
{
    pid_t pid;
    FD *fd;
    struct pidInfo_link* next;
}pidLink;
```

- I decide to use linked list to store information because I don't know the number of the valid accessible pid, linked list is useful when I traverse every nodes, and it is easy to free each

node to avoid memory leak

- use `DIR *dir = opendir(dirname)` to open a directory
- use `struct dirent *entry` `entry = readdir(dir)` to go through each element under that directory `dir`, `entry->d_name` is the elements name.
- In order to check accessible pid, I go to `/proc` directory, the processing pids are in that directory, use `struct dirent *entry` to get directory name `entry->d_name` which is the pid if the name of directory can be a valid integer, check `access(/proc/[pid]/fd, R_OK)==0` to find the accessible pid.
- In each pid, `/proc/[pid]/fd` in this directory, it contains the FD under the [pid], use the function `readlink` in `<unistd.h>` to get the corresponding filename corresponds to FD number, use `struct stat sb lstat(/proc/[pid]/fd/[FD]` to get each fd's inode, which can be find by `sb.st_ino`

Once the information is stored in the linked list, it is easy to implement the desired output.

- For `--per-process --systemWide --Vnodes --composite --threshold=X` traverse the linked linked list, find corresponding required information, then printed them out
- For `--output_TXT` use `fprintf` to store `composite table` in ASCII format into `compositeTable.txt` remember to `fopen("compositeTable.txt", "w")` and `fclose(FILE *f)`
- For `--output_binary` use `fwrite` to store `composite table` in binary format into `compositeTable.bin` remember to `fopen("compositeTable.bin", "wb")` and `fclose(FILE *f)`

Overview of the function

Function	Description
`typedef struct pidInfo`	it stores the FD, filename, inode under a certain pid, it is a linked list, I decide its another name as FD
`typedef struct pidInfo_link`	it is a linked list, it contains the pid, and the pidInfo, the pidInfo stores the FD, filename, inode under a certain pid
`pidLink* newpid(pid_t pid)`	Allocate a space for a new pid, with the new node->pid = pid
`FD *newfd()`	Allocate a space for a new FD, set the content of that FD node in default value
`bool validInt(char *s)`	return True if the s can be convert to a integer,

Function	Description
	False otherwise
`pidLink* findPid(pid_t pid, pidLink* head)`	find the node that node->pid = pid
`void deleteFD(pidLink *node)`	used to free each `FD *fd` under the node
`void deletePID(pidLink *head)`	used to free the whole pidLink
`int countFD(FD *node)`	return how many nodes in the linked list node
`pidLink *insertFD(pid_t pid, pidLink *node)`	used to insert the corresponding FD information corresponds to the specific pid to the node
`pidLink *insertpid(pid_t pid, pidLink *head)`	used to insert pidLink node with node->pid = pid, insert that node into head
`void perprocess(pid_t pid, pidLink *head, bool flag)`	check `--per-process`, print corresponding output if command line `--per-process` is shown, if the pid != -1, then print output info corresponds to the specific pid
`void systemWide(pid_t pid, pidLink *head, bool flag)`	check `--systemWide`, print corresponding output if command line `--systemWide` is shown, if the pid != -1, then print output info corresponds to the specific pid
`void vnode(pid_t pid, pidLink *head, bool flag)`	check `--Vnodes`, print corresponding output if command line `--Vnodes` is shown, if the pid != -1, then print output info corresponds to the specific pid
`void composite(pid_t pid, pidLink *head,bool flag)`	check `--composite`, print corresponding output if command line `--composite` is shown, if the pid != -1, then print output info corresponds to the specific pid
`void txtoutput(pid_t pid, pidLink *head, bool flag)`	check `--output_TXT`,save the "composite" table in *text (ASCII)* into a file named `compositeTable.txt` if command line `--output_TXT` is shown, if the pid != -1, the composite table is corresponding to that specific pid, save it into `compositeTable.txt`
`void binaryoutput(pid_t pid, pidLink *head,bool flag)`	check `--output_binary`,save the "composite" table in binary format into a file named `compositeTable.bin` if command line `--output_binary` is shown, if the pid != -1, the composite table is corresponding to that specific pid, save it into `compositeTable.txt`
`void threshold(int thresholdNum,`	print processes which have a number of FD

Function	Description
`pidLink *head,bool flag)`	assigned larger than `_X_`, in PID(#FD) format
`pid_t checkFlag(int argc, char** argv, bool* processflag, bool* systemWideflag, bool* Vnodesflag, bool* compositeflag, bool* thresholdflag, int* thresholdNum, bool* output_TXTflag, bool* output_binaryflag)`	Check the command line arguments, if flag is found, make the flag be true

Running Code

Command Line

- `--per-process` :indicates that only the process FD table will be displayed
- `--systemWide`, indicates that only the system-wide FD table will be displayed
- `--Vnodes`, indicates that the Vnodes FD table will be displayed
- `--composite`, indicates that only the composed table will be displayed
- `--threshold=X`, where `X` denotes an integer, indicating that processes which have a number of FD assigned larger than `X` should be flagged in the output.
- `--output_TXT`, when the flag is used the program will save the "composite" table in *text (ASCII)* into a file named `compositeTable.txt`
- `--output_binary`; when the flag is used the program will save the "composite" table in *binary format* into a file named `compositeTable.bin`.
-
  - positional argument: only one positional argument indicating a particular process id number (PID), if not specified the program will attempt to process all the currently running processes for the user executing the program

By default

- the program will display the composite table, i.e. same effect as having used the `--composite` flag

Sample running

- `gcc -Wall A2.c -o showFDtables` `./showFDtables --composite`

```
./showFDtables  --composite

        PID     FD      Filename                Inode
       ================================================
 0      118734  0       /dev/null               5
 1      118734  1       socket:[636647]         636647
 2      118734  2       socket:[636647]         636647
 3      118734  3       socket:[641257]         641257
 4      118734  4       anon_inode:[eventpoll]  12533
 5      118734  5       anon_inode:[signalfd]   12533
 6      118734  6       anon_inode:inotify      12533
 7      118734  7       /sys/fs/cgroup/user.slice/user-17524891.slice/user@17524891.service     35275
 8      118734  8       anon_inode:[timerfd]    12533
 9      118734  9       anon_inode:[eventpoll]  12533
 10     118734  10      /proc/118734/mountinfo  625599
 11     118734  11      anon_inode:inotify      12533
 12     118734  12      anon_inode:inotify      12533
 13     118734  13      anon_inode:inotify      12533
 14     118734  14      /proc/swaps     4026532089
 ...
 291    121210  6       socket:[653518]         653518
 292    121210  7       socket:[652701]         652701
 293    121210  9       /dev/ptmx       87
 294    121302  0       /dev/pts/3      6
 295    121302  1       /dev/pts/3      6
 296    121302  2       /dev/pts/3      6
 297    121302  255     /dev/pts/3      6
 298    123306  0       /dev/pts/3      6
 299    123306  1       /cmsfaculty/marcelo/project/desrip.id  4598386659
 300    123306  2       /dev/pts/3      6
 301    123306  3       /proc   1
 302    123306  4       /proc/123306/fd         671451
        ========================================
```

  It is same as `gcc -Wall A2.c -o showFDtables` `./showFDtables` which have the same effect as `--composite`

- `gcc -Wall A2.c -o showFDtables` `./showFDtables 118743` This will output the following 4 photos. Assume 118743 is a valid and accessible PID

```
./showFDtables 118743

        PID     FD
       ============
        118743  0
        118743  1
        118743  2
        118743  3
        118743  4
        118743  5
        118743  6
        118743  7
        118743  8
        118743  9
        118743  10
        118743  11
        118743  12
        118743  13
        118743  14
        118743  15
        118743  16
        118743  17
        118743  18
        118743  19
        118743  20
        118743  21
        118743  22
        118743  23
        118743  24
        118743  25
       ===========
```

  The above one is same as `./showFDtables 118743 --per-process`

```
 PID     FD      Filename
========================================
 118743  0       /dev/null
 118743  1       socket:[638144]
 118743  2       socket:[638144]
 118743  3       socket:[625628]
 118743  4       pipe:[639499]
 118743  5       pipe:[639499]
 118743  6       /memfd:pulseaudio (deleted)
 118743  7       pipe:[639500]
 118743  8       pipe:[639500]
 118743  9       socket:[642071]
 118743  10      /Users/marcelo/.config/pulse/27303920a74b-device-volumes.tdb
 118743  11      /Users/marcelo/.config/pulse/27303920a74b-stream-volumes.tdb
 118743  12      /Users/marcelo/.config/pulse/27303920a74b-card-database.tdb
 118743  13      anon_inode:inotify
 118743  14      socket:[642069]
 118743  15      socket:[642070]
 118743  16      anon_inode:[eventfd]
 118743  17      anon_inode:[eventfd]
 118743  18      anon_inode:[eventfd]
 118743  19      anon_inode:[eventfd]
 118743  20      anon_inode:inotify
 118743  21      anon_inode:[eventfd]
 118743  22      anon_inode:[eventfd]
 118743  23      anon_inode:[eventfd]
 118743  24      anon_inode:[eventfd]
 118743  25      socket:[642073]
========================================
```

The above one is same as `./showFDtables 118743 --systemWide`

```
   FD            Inode
========================================
   0              5
   1              638144
   2              638144
   3              625628
   4              639499
   5              639499
   6              6342
   7              639500
   8              639500
   9              642071
   10             4551371476
   11             4551371477
   12             4551371478
   13             12533
   14             642069
   15             642070
   16             12533
   17             12533
   18             12533
   19             12533
   20             12533
   21             12533
   22             12533
   23             12533
   24             12533
   25             642073
========================================
```

The above one is same as `./showFDtables 118743 --Vnodes`

```
 PID     FD       Filename       Inode
=======================================
 118743  0       /dev/null       5
 118743  1       socket:[638144]         638144
 118743  2       socket:[638144]         638144
 118743  3       socket:[625628]         625628
 118743  4       pipe:[639499]   639499
 118743  5       pipe:[639499]   639499
 118743  6       /memfd:pulseaudio (deleted)     6342
 118743  7       pipe:[639500]   639500
 118743  8       pipe:[639500]   639500
 118743  9       socket:[642071]         642071
 118743  10      /Users/marcelo/.config/pulse/27303920a74b-device-volumes.tdb  4551371476
 118743  11      /Users/marcelo/.config/pulse/27303920a74b-stream-volumes.tdb  4551371477
 118743  12      /Users/marcelo/.config/pulse/27303920a74b-card-database.tdb   4551371478
 118743  13      anon_inode:inotify      12533
 118743  14      socket:[642069]         642069
 118743  15      socket:[642070]         642070
 118743  16      anon_inode:[eventfd]    12533
 118743  17      anon_inode:[eventfd]    12533
 118743  18      anon_inode:[eventfd]    12533
 118743  19      anon_inode:[eventfd]    12533
 118743  20      anon_inode:inotify      12533
 118743  21      anon_inode:[eventfd]    12533
 118743  22      anon_inode:[eventfd]    12533
 118743  23      anon_inode:[eventfd]    12533
 118743  24      anon_inode:[eventfd]    12533
 118743  25      socket:[642073]         642073
 =======================================
```

The above one is same as `./showFDtables 118743 --composite`

- `gcc -Wall A2.c -o showFDtables` `./showFDtables threshold=20`

```
         PID     FD      Filename               Inode
=============================================
0        118734  0       /dev/null              5
1        118734  1       socket:[636647]        636647
2        118734  2       socket:[636647]        636647
3        118734  3       socket:[641257]        641257
4        118734  4       anon_inode:[eventpoll] 12533
5        118734  5       anon_inode:[signalfd]  12533
6        118734  6       anon_inode:inotify     12533
7        118734  7       /sys/fs/cgroup/user.slice/user-17524891.slice/user@17524891.service     35275
8        118734  8       anon_inode:[timerfd]   12533
9        118734  9       anon_inode:[eventpoll] 12533
10       118734  10      /proc/118734/mountinfo 625599
11       118734  11      anon_inode:inotify     12533
12       118734  12      anon_inode:inotify     12533
13       118734  13      anon_inode:inotify     12533
14       118734  14      /proc/swaps    4026532089
...
291      121210  6       socket:[653518]        653518
292      121210  7       socket:[652701]        652701
293      121210  9       /dev/ptmx      87
294      121302  0       /dev/pts/3     6
295      121302  1       /dev/pts/3     6
296      121302  2       /dev/pts/3     6
297      121302  255     /dev/pts/3     6
298      123306  0       /dev/pts/3     6
299      123306  1       /cmsfaculty/marcelo/project/desrip.id  4598386659
300      123306  2       /dev/pts/3     6
301      123306  3       /proc   1
302      123306  4       /proc/123306/fd        671451
         =======================================
```

```
## Offending processes:
118743 (25), 234678 (456), ...
```

- `gcc -Wall A2.c -o showFDtables` `./showFDtables --output_TXT --output_binary`
  when these flags are used the program will save the "composite" table in *text (ASCII)* or *binary format* into a file
  named `compositeTable.txt` or `compositeTable.bin` respectively.

Also you can try any combination of those command line, but notice, the positional argument has to be a valid and accessible PID.
EX) `./showFDtables --per-process --systemWide --Vnode --composite`
`./showFDtables --per-process --systemWide --Vnode --composite 118743` Assume 118743 is a valid and accessible PID

`./showFDtables --per-process --systemWide --Vnode --composite 118743 --output_TXT --output_binary`

Using the Makefile to implement my code

`make help` : will show the help section

`make` : will run `./showFDtables`

`make clean` : will Remove generated file

More information can be found in `make help`

Observation

`time ./showFDtables --output_TXT --output_binary`

```
lianhao@Lianhao:~/projects/CSCB09/A2$ time ./showFDtables --output_TXT --output_binary

real    0m0.002s
user    0m0.000s
sys     0m0.000s
```

real is the time from start to finish of the call
user is the amount CPU time spent in user mode
sys is the amount of CPU time spent in kernel mode.