Question - 2 Readme file

First , we take input from stdin and assign all required input into four different variable. Command variable consist of commands like word , dir , date , cmd , exit. Same with option and filename. If given command is word then we make fork() system call and in child process we do our word computation . In child process , we check if option == -n, -d , and no option and work as per their respective option . We use execv() system call to call function word_file and pass arguments {“./word_file”,”-n”,filename) to the word_file. In word_file , we check argv[1] which is our option argument and if argv[1] == ‘-n’ , we run program for that option and if argv[1] == ‘-d’ , we run program for that option and if argv[1] == ‘0’ which means no option passed . To calculate number of words , we make function which we take every ch from the input.txt file and check if there is any ‘ ‘ or ‘/n’ and if there then increment word_cnt. It also checks if there is any continuous space or \n then don't count it as a word. We also have fileCheck function which will check whether the given file exists or not . Now for command dir, it do the same fork() work and

Also checks same for the option . When we dont provide any option it simpy make dir using mkdir() function . This function return -1 if directory already exists and if it exists it throws error and if not it makes dir using the same function . If we provide -r option , it do the same but additionally we make an array cmd and with sprintf formatting function we assign rm -rf in this cmd array with the directory name. This cmd will delete provided directory whether the directory is empty or not . After that we make directory we the given directory name we want to make.

Option -v is simple it just printing everything we are doing . Now for date command, we do the same fork() , execv() thing . We use stat struct . This stat struct stores the time of modification , when access first , and many other things of a file. So we use this structure to get the time of the modification of a file. If there is no option provided , we simply check existence of file and calls IST_TIME function which makes stat variable names attr and give that attr the location of the file and simply print the time in IST using ctime by getting time by attr.st_mtime as the struct stat stores variable st_mtime. Same with -R option , only difference is we are printing time in RFC 5322 format. For this we use strftime . For option -d , we take one more string variable , but we are **ASSUMING** that we can take only yesterday or today value . This we calculate last modification time of yesterday and today. Format is : **date -d yesterday test1.txt** . not **‘yesterday’** .

Assumption - We already provided to input files. -r in dir command will delete directory already exists and then create new directory as mentioned in question .