Author: Jenna Wolf

Program: JennaCFile

Class: C and the Posix environment

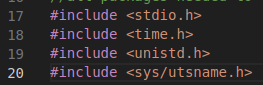
Assignment: Project 1 Linux C Program

Version: 3.16.24

Revision: 1

How To Use JennaCFile program

Purpose: The purpose of this program is to fit the guidelines for the project 1 Linux system c program assignment. This program works to access and display system information to the user. It also uses error checking to make sure proper inputs are made and display error messages when information retrieval fails.

Packages:

#include <stdio.h> - this package allows a c program to be made and ran. It allows inputs and outputs to be made in a c program.

#include <time.h> - allows time structure, which allows the time and date to be displayed

#include <unistd.h> - the package allows many different functions used throughout the program, such as the sleep function and the gethostname function.

#include <sys/utsname.h> - this package allows the program to find out system information, such as the kernel version and the release information

Variables:



The choice variable. This is the variable that the user inputs to. It is a integer, so it will only take numbers from the user. The program goes based on the input this variable receives.



The timeNow variable and the tm variable are the variables used to hold and find the current date and time. The time\_t and struct tm are used for time-related operations. TimeNow holds the current date and time, while tm structures it into a easily readable format

The host variable is a character array of size 256. This is used to hold the hostname like a string. When the gethostname function is called, this variable is set to the host name.



The data variable is used to find and hold system information such as the release information and kernel version. Struct utsname is used to access system information, and the uname function actually sets to data variable to that system information.



The nop variable is used to hold the number of processors/cores. The long type is used here because the value returned may need more memory space than an integer can hold. Sysconf is a function used to get system information. In this case, it is setting nop to the current number of cpus online using the parameter \_SC\_NPROCESSORS\_ONLN.

The totalmem varaible is used to hold the total amount of memory (RAM). This is a long type since this is a larger number, and we are again using the sysconf function since it has access to system information. We use the parameters \_SC\_PHYS\_PAGES and \_SC\_PAGE\_SIZE because when the amounts they return are multiplied together, they give us the total amount of physical memory on the system. PHYS\_PAGES gets the total number of physical memory pages on the system, and PAGE\_SIZE gets the size of a memory page in bytes. It is then divided by 1024\*1024 as this is the amount of bytes in a megabyte, which is the value we are storing memory as.

**Code segment:**

The beginning of the code (lines 43-44) says a simple welcome statement to the user. It they sleeps for one second so the screen is not overwhelmed with statements right away.



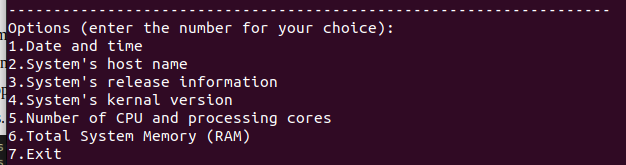
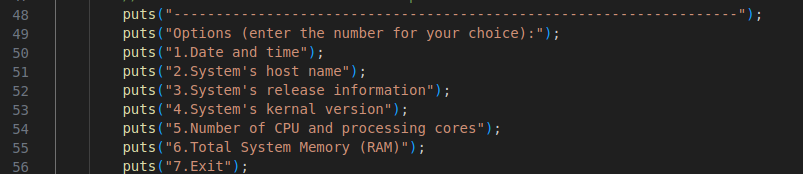


The program then enters a do while loop. This loop starts on line 46 and ends on line 137. This is the main part of the program, as all operations go on in here. The loop exits when 7 is entered for the choice variable.

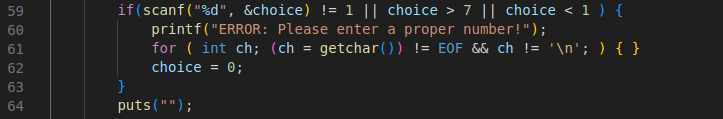




We then reach the menu part of the program (lines 48-56). This is the section that tells the user what to do. It lists the options the user has, along with a number for each option so the user knows what to input. There are 7 options for the user to choose from (described from lines 50-56). This it output every time the code loops.



The input section of the code (lines 59-64) is where the user gets to interact with the code. This is made into an if statement to check what the user is entering, This is used to make sure the user enters an integer, as if they do not, the if statement is entered. The if statement is also entered when an integer less that 1 or greater that 7 is entered. When the if statement is gone through, an error message is sent to the user first (line 60). This simply asks the user to input a proper input next time. After that, a for statement is gone through (61). This for loop is used to consume and clear out input if a character or string was entered. It goes through all the characters until it reaches a EOF or end character. Choice is then set to 0 (line 62) as a way of resting the input. Lastly, the puts lien (64) adds a blank line to the output of the program.

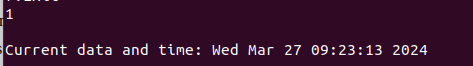
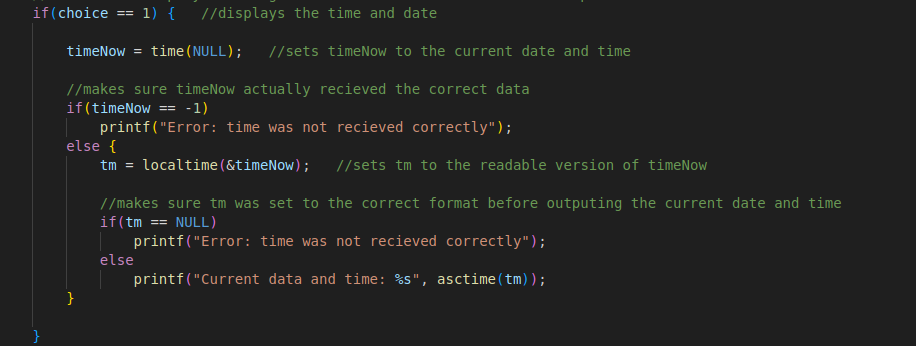


6

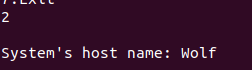
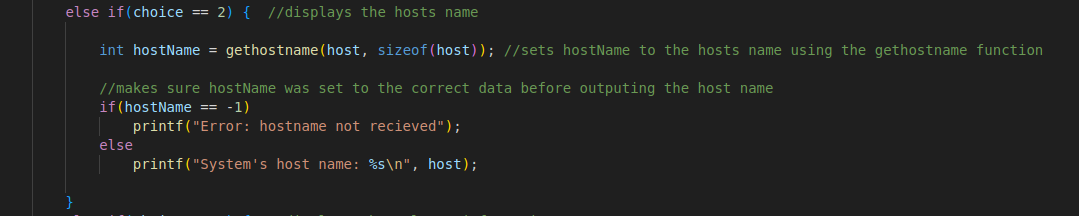




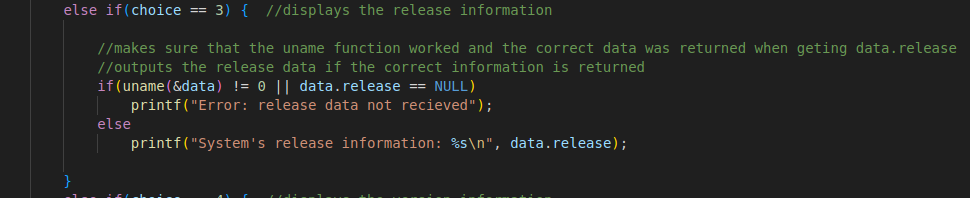
Many if statements makes up the rest of this code. The first one (line 67) sees if the user entered 1 as their choice. This is the code for displaying the date and time (lines 68-84). The first piece of code (67) sets timeNow to the current time and date using the time function. We then reach another if statement (lines 72-82) which checks to make sure that timeNow was actually set and did not return an error. If it did not error out, then we go to the line assigns tm using the localtime function (line 75). This function takes the timeNow data and sets it to a readable format. We then reach our last if statement (line 78-81), which makes sure that tm was not set to Null. This is just making sure that tm was actually set to something before it is output. We then reach a print statement that uses the asctime function to output the tm data as a readable human string.



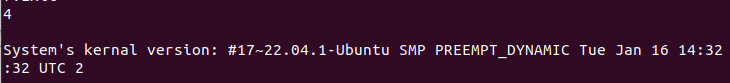
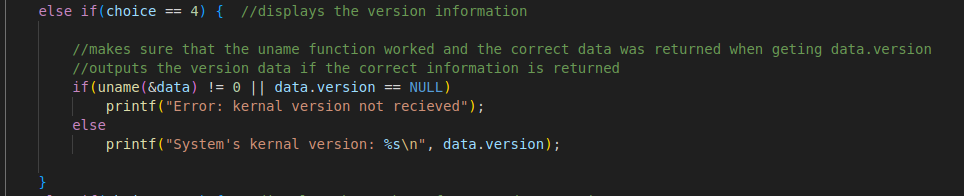
The next if statement is used for the host name. This line (line 85) asks if the user entered 2 as their choice. If they did, we enter the if statement (lines 86-95). We start by creating a hostName variable (line 87) that uses the gethostname function. The parameters for this function uses the host character array along with the size of this variable to get and hold the host name. We then hit a if statement (lines 90-93) that make sure that hostName was actually set and did not return a error value. If this if statement is passed, then the host name is displayed as a string on the screen.

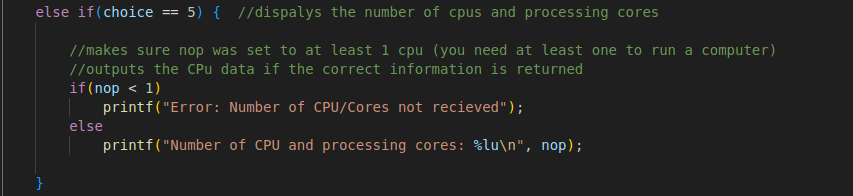


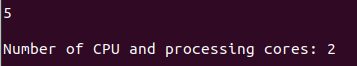
The next if statement is used for the release information. This line (line 96) asks if the user entered 3 as their choice. If they did, we enter the if statement (lines 97-105). This chunk of code is simply made up of an if statement (lines 100-103) that makes sure of two things. It makes sure that the uname function with the data parameter is set to 0, as a 0 return means that the function was successful. and it makes sure that the data.release value is not returning a null value. If both of these are passed, then the release information is output to the user as a string.



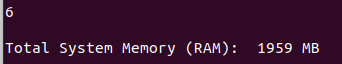
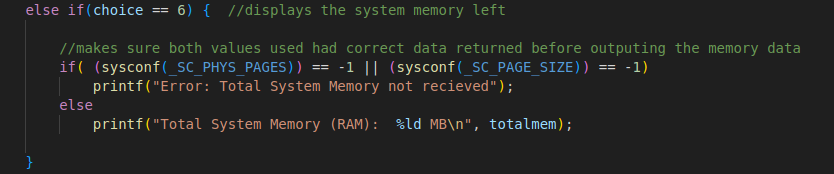
The next if statement is used for the kernel information. This line (line 106) asks if the user entered 4 as their choice. If they did, we enter the if statement (lines 107-115). This chunk of code is simply made up of an if statement (lines 110-113) that makes sure of two things, just like the previous chunk of code. It makes sure that the uname function with the data parameter is set to 0, as a 0 return means that the function was successful. and it makes sure that the data.version value is not returning a null value. If both of these are passed, then the Kernal version information is output to the user as a string.



The next if statement is used for the numbers of cpu’s/cores. This line (line 116) asks if the user entered 5 as their choice. If they did, we enter the if statement (lines 117-125). This chunk of code is simply made up of an if statement (lines 120-123). This if statement makes sure that nop (number of processors) is at least one, since a computer needs at least one cpu to work. If this condition is passed, then a string is output with the number of cpus.



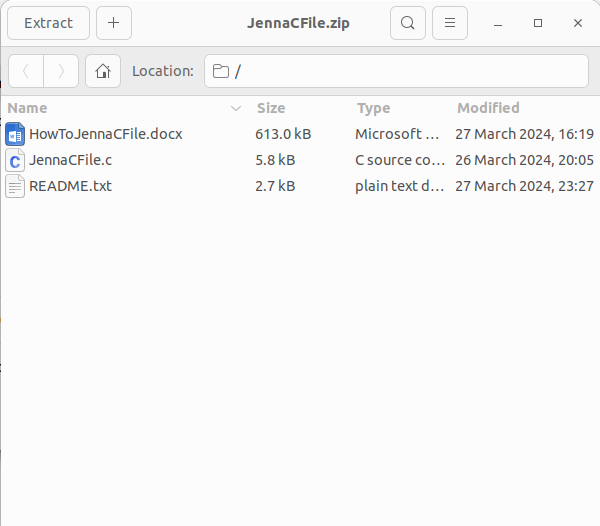
The last if statement is used for the amount of memory left. This line (126) asks if the user entered a 6 as their choice. If they did, we enter the statement (lines 127-134). This chunk of code is simply made up of an if statement (lines 110-113) that makes sure of two things (again). It makes sure that the sysconf commands for both pages and page size do not return a negative one (meaning they don’t return an error value). If this condition is passed, then the total memory is output in a string.



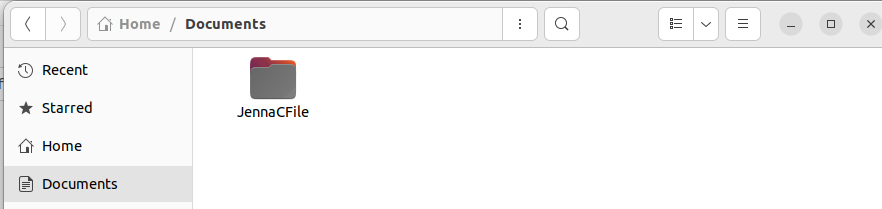
This ends the if statements. We then reach the end of the while loop (lines 135-136). These lines simply add a space to the screen for presentation purposes, and then sleeps for one second before the loop repeats or the loop is exited. The very last line of the program (line 139) is return 0, which ends the codes execution.

**How to run:**

1. Download the zip file. The zip file should look like the following and have three files inside: a C file, a read me, and a how to (you most likely have done this step if you are reading this.)

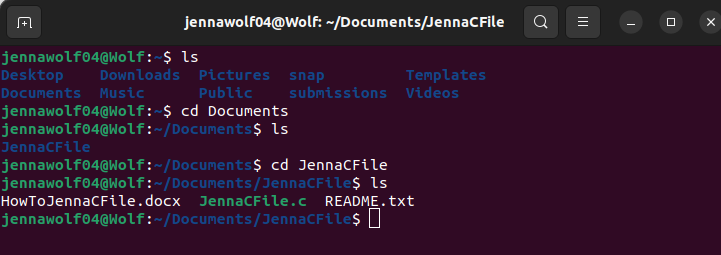


2. Unzip the file into the directory/fouler you wish to run the program from. Please note where you put this file, as you will need it to run the program. Documents is suggested, but not required. You can then delete the zip file after this, as you should have a folder with the name of the zip file (before deleting, enter your new folder to make sure all files are there).

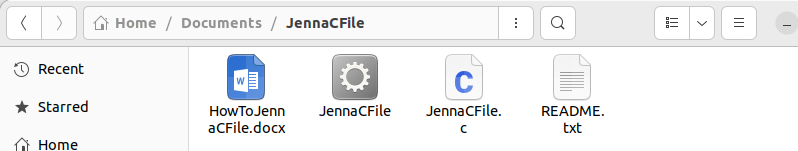


3. open up your terminal.

4. use the cd and ls commands to enter the directory/folder with the program. Cd with a folder/directory name will place you in the folder/directory you pick. Ls tells you where you can go from your current directory. When you are in the correct directory, you should be able to see the runnable c program JennaCFile.cpp . It should look like the picture below.



5. You then need to mod your program to work. Simply type in the command “ gcc JennaCFile.c -o JennaCFile “ to do so. When you hit enter on this command, you should not get an error on screen, and if you check your folder, you will see a new file in there. This is the runnable version of the program.



6. Finally, type in “ ./JennaCFile “ to run the program. From here, you can follow the prompts on screen to use the program. Keep in mind that only number inputs will work with this program.

