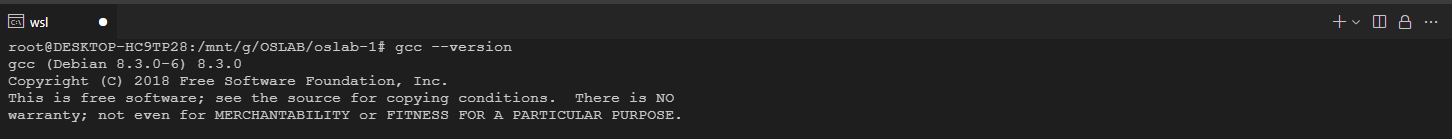
OPERATING SYSTEMS LAB-3 WEEK-4

Author: Muhammad Ahsan 1912310 BSCS 5F

1. TASK 1: INSTALL C-LANGAUGE COMPILER IN LINUX OS

-- sudo apt install gcc [INSTALLATION OF GCC FOR C]

-- gcc --version [To Check Version of gcc compiler]



2. TASK 2: WRITE A PROGRAM IN C TO PRINT "HELLO WORLD"

-- nano hello.c [To Create C file to write hello world]

-- #include <stdio.h>

void main(){

printf("Hello World");

}

-- CTRL+X [save quit to terminal and save file in nano]

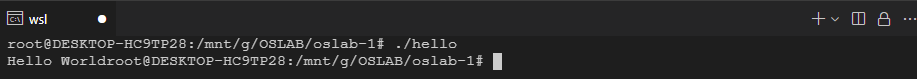
-- gcc hello.c -o hello [Compile C file with Output executable using this command]

-- ./hello [Execute the file using this command]

-- Now if we make changes we will just compile it and overwrite it with the same name

-- gcc hello.c -o hello

-- ./hello



3. TASK 3: TAKE INPUT FROM USER AND PRINT IT ON THE SCREEN

-- nano input.c [To Create C file to take input from user and print it on screen]

-- #include <stdio.h>

void main(){

int number;

printf("ENTER NUMBER INPUT:");

scanf("%d",&number);

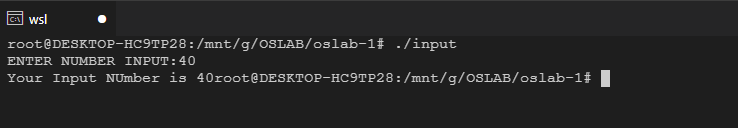
printf("Your Input Number is %d",number);

}

-- CTRL+X [save quit to terminal and save file in nano]

-- gcc input.c -o input [Compile C file with Output executable using this command]

-- ./input [Execute the file using this command]



4. TASK 4: TAKE INPUT NUMBER FROM USER AND PRINT IT'S TABLE ON THE SCREEN (TILL 10)

-- nano table.c

[To Create C file to take input number from user and print it's table (TILL 10) on screen]

-- #include <stdio.h>

void main(){

int number;

printf("ENTER NUMBER INPUT:");

scanf("%d",&number);

for(int i=1;i<=10;i++){

printf("%d\*%d=%d",i,number,i\*number);

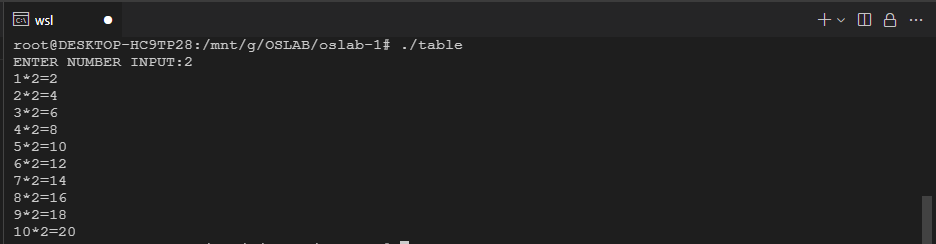
}

}

-- CTRL+X [save quit to terminal and save file in nano]

-- gcc table.c -o table [Compile C file with Output executable using this command]

-- ./table [Execute the file using this command]



5. TASK 5: CALL FORK IN A PROGRAM TO CREATE SUB PROCESS AND COMPARE IT'S ID WITH PARENT AND CHILD

-- nano forkProgram.c

-- #include <stdio.h>

#include <unistd.h>

void main(){

int output=fork();

if(output==0){

printf("\n Called From Child \n");

}

else{

printf(" \n Called From Parent");

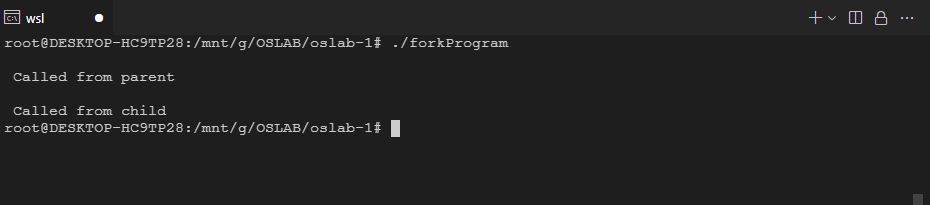
}

}

-- CTRL+X [save quit to terminal and save file in nano]

-- gcc forkProgram.c -o forkProgram [Compile C file with Output executable using this command]

-- ./forkProgram [Execute the file using this command]



6. TASK 6: CALL FORK IN A PROGRAM TO CREATE SUB PROCESS AND COMPARE IT'S ID WITH PARENT AND CHILD AND OUTPUT IT'S PROCESS ID

-- nano forkProgram2.c

-- #include <stdio.h>

#include <unistd.h>

pid\_t pid;

void main(){

int output=fork();

pid=getpid();

if(output==0){

printf("\n Called From Child With Process ID %d \n",pid);

}

else{

printf("\n Called From Parent With Process ID %d \n",pid);

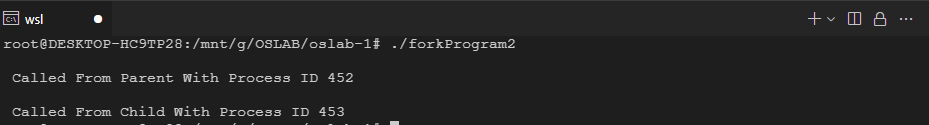
}

}

-- CTRL+X [save quit to terminal and save file in nano]

-- gcc forkProgram2.c -o forkProgram2 [Compile C file with Output executable using this command]

-- ./forkProgram2 [Execute the file using this command]



7. TASK 7: CALL FORK IN A PROGRAM TO CREATE SUB PROCESS (2 FORK) AND COMPARE IT'S ID WITH PARENT AND CHILD AND OUTPUT IT'S PROCESS ID

-- nano forkProgram3.c

-- #include <stdio.h>

#include <unistd.h>

pid\_t pid;

void main(){

int output=fork();

fork();

fork();

pid=getpid();

if(output==0){

printf("\n Called From Child With Process ID %d \n",pid);

}

else{

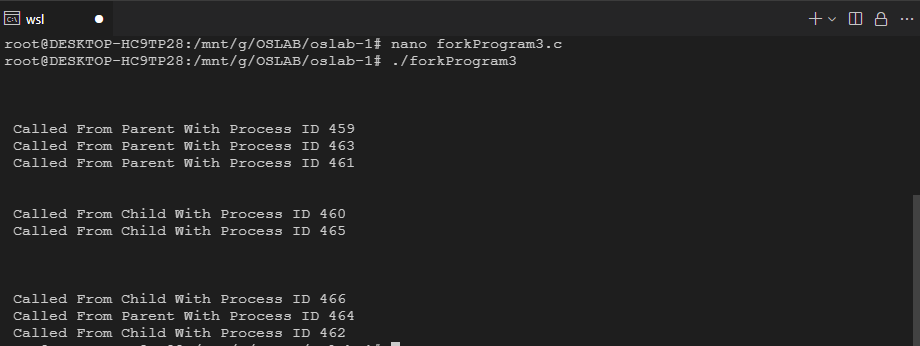
printf("\n Called From Parent With Process ID %d \n",pid);

}

}

-- CTRL+X [save quit to terminal and save file in nano]

-- gcc forkProgram3.c -o forkProgram3 [Compile C file with Output executable using this command]

 -- ./forkProgram3 [Execute the file using this command]