

```
GNU nano 7.2 process_creation.c *
#include <stdio.h>
#include <unistd.h>

int main() {
    pid_t pid = fork();
    if (pid == 0) {
        printf("This is the child process. PID: %d\n", getpid());
    } else if (pid > 0) {
        printf("This is the parent process. PID: %d\n", getpid());
    } else {
        printf("Fork failed!\n");
    }
    return 0;
}
```

```
GNU nano 7.2 file1.c
void hello() {
    printf("Hello from file1!\n");
}
```

```
GNU nano 7.2 file2.c
void hello();

int main() {
    hello();
    return 0;
}
```

```
GNU nano 7.2 simple_program.c
#include <stdio.h>

int main() {
    printf("This is a simple program.\n");
    return 0;
}
```

```
GNU nano 7.2 Makefile *
# Define the compiler
CC = gcc

# Set compilation flags
CFLAGS = -Wall -g

# Default target: compile all
all: process_creation output_program simple_program

# Compile process_creation.c
process_creation: process_creation.c
    $(CC) $(CFLAGS) process_creation.c -o process_creation

# Compile file1.c and file2.c together
output_program: file1.c file2.c
    $(CC) $(CFLAGS) file1.c file2.c -o output_program

# Compile simple_program.c
simple_program: simple_program.c
    $(CC) $(CFLAGS) simple_program.c -o simple_program

# Remove compiled binaries
clean:
    rm -f process_creation output_program simple_program

# Run all programs
```

^G Help	^O Write Out	^W Where Is	^K Cut	^T Execute	^C Location	M-U Undo	M-A Set Mark
^X Exit	^R Read File	^I Replace	^U Paste	^J Justify	^_ Go To Line	M-E Redo	M-G Copy

```
noreen8@DESKTOP-1UV6SC4 x + v
GNU nano 7.2 Makefile *
all: process_creation output_program simple_program

# Compile process_creation.c
process_creation: process_creation.c
$(CC) $(CFLAGS) process_creation.c -o process_creation

# Compile file1.c and file2.c together
output_program: file1.c file2.c
$(CC) $(CFLAGS) file1.c file2.c -o output_program

# Compile simple_program.c
simple_program: simple_program.c
$(CC) $(CFLAGS) simple_program.c -o simple_program

# Remove compiled binaries
clean:
rm -f process_creation output_program simple_program

# Run all programs
run: all
./process_creation
./output_program
./simple_program

.PHONY: all clean run

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo      M-A Set Mark
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^_ Go To Line  M-E Redo      M-6 Copy
```

```
noreen8@DESKTOP-1UV6SC4 x + v
GNU nano 7.2 README.md
# CSE233 Lab-5: Process Management and Basic Forks

## Overview
This project demonstrates basic process management in C using fork, multi-file compilation, and the usage of Makefile f

## Contents
- `process_creation.c` - Demonstrates fork and process creation
- `file1.c`/`file2.c` - Multi-file compilation and linking demo
- `simple_program.c` - Loader example
- `Makefile` - Automates build and run commands
- `LICENSE` - Project license (MIT)
- `.txt` file - Answers and explanations
- Screenshots - Proof of code compilation and execution

## How to Build
Make sure `gcc` and `make` are installed.

## How to Run
make
make all

## How to Run
make run
```

```
## How to Run
make run

## How to Clean Up
make clean

## License
Distributed under the MIT License. See LICENSE for details.

## Author
Noreen Osama
23102386

## Course
Operating Systems - CSE233, Alamein University
```

```
noreen8@DESKTOP-1UV6SC4  ×  +  v
noreen8@DESKTOP-1UV6SC4:~$ nano process_creation.c
noreen8@DESKTOP-1UV6SC4:~$ sudo apt update
sudo apt install gcc
[sudo] password for noreen8:

noreen8@DESKTOP-1UV6SC4:~$ gcc process_creation.c -o process_creation
noreen8@DESKTOP-1UV6SC4:~$ ./process_creation
This is the parent process. PID: 1171
This is the child process. PID: 1172

noreen8@DESKTOP-1UV6SC4:~/ass2$ nano file1.c
noreen8@DESKTOP-1UV6SC4:~/ass2$ nano file2.c
noreen8@DESKTOP-1UV6SC4:~/ass2$ gcc file1.c file2.c -o output_program.c
noreen8@DESKTOP-1UV6SC4:~/ass2$ ./output_program
-bash: ./output_program: No such file or directory
noreen8@DESKTOP-1UV6SC4:~/ass2$ gcc file1.c file2.c -o output_program
noreen8@DESKTOP-1UV6SC4:~/ass2$ ./output_program
Hello from file1!
noreen8@DESKTOP-1UV6SC4:~/ass2$ |

noreen8@DESKTOP-1UV6SC4:~$ nano simple_program.c
noreen8@DESKTOP-1UV6SC4:~$ gcc simple_program.c -o simple_program
noreen8@DESKTOP-1UV6SC4:~$ ./simple_program
This is a simple program.

noreen8@DESKTOP-1UV6SC4:~$ nano Makefile
noreen8@DESKTOP-1UV6SC4:~$ make clean
rm -f process_creation output_program simple_program
noreen8@DESKTOP-1UV6SC4:~$ make all
gcc -Wall -g process_creation.c -o process_creation
gcc -Wall -g file1.c file2.c -o output_program
gcc -Wall -g simple_program.c -o simple_program
noreen8@DESKTOP-1UV6SC4:~$ make run
./process_creation
This is the parent process. PID: 2156
This is the child process. PID: 2157
./output_program
Output of file1!
./simple_program
This is a simple program.
noreen8@DESKTOP-1UV6SC4:~$ |
```