**Họ và Tên: Phạm Thị Hồng Linh**

**MSSV: B1809365**

Tiện ích Make

# Tạo thư mục LapTrinhC: mkdir LapTrinhC Bài 0 - hello.c

- Tạo thư mục **Bai0** trong thư mục LapTrinhC: **mkdir Bai0**

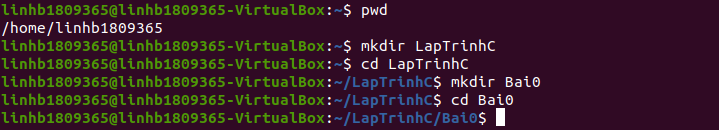
* Mở chương trình soạn thảo văn bản
* Mở slide bài giảng chủ đề Lập trình C/Linux
  + Di chuyển đến slide số 5
  + Copy nội dung của chương trình **hello.c** ở slide số 5
* Dán nội dung của chương trình vào trình soạn thảo văn bản

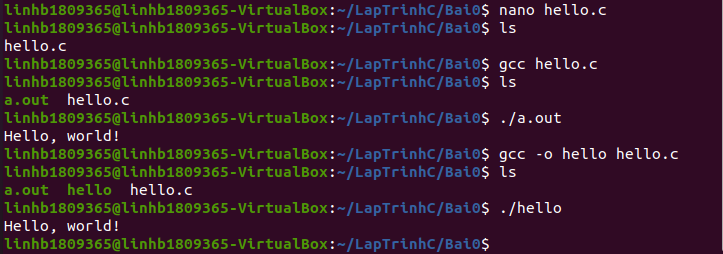
- Lưu tập tin với tên là ***hello.c*** trong thư mục Bai0

* Dịch chương trình
  + Mở một terminal
  + Chuyển vào thư mục Bai0
  + Đánh lệnh **ls** để xem nội dung thư mục Bai0
  + Đánh lệnh sau để dịch chương trình hello.c: **gcc hello.c**
  + Đánh lệnh **ls** để xem nội dung thư mục Bai0

Ta thấy có tập tin *a.out* mới được tạo ra từ lệnh biên dịch phía trên

* + Chạy chương trình a.out bằng lệnh sau: **./a.out**
  + Sử dụng tiện tùy chọn -o để dịch lại chương trình hello.c thành chương trình thực thi có tên là hello thay vì là a.out: **gcc -o hello hello.c**
  + Đánh lệnh **ls** để xem nội dung thư mục Bai0
  + Thực thi chương trình hello: **./hello**





# Bài 1:Viết hàm int cong (int a,int b) trả về kết quả là phép toán cộng 2 số a và b

* Tạo thư mục **Bai1** để lưu các tập tin chương trình
* Soạn thảo tập tin có tên là **cong.c** và lưu vào thư mục Bai1 có nội dung như sau:

#include<stdio.h> int cong(int a, int b)

{

int ketqua; ketqua = a + b; return ketqua;

}

* Đánh lệnh sau để biên dịch tập tin cong.c thành chương trình thực thi cong

## gcc -o cong cong.c

* Bạn sẽ gặp thông báo lỗi sau:

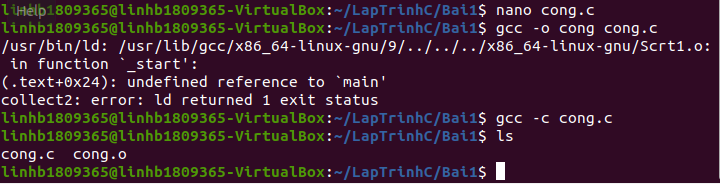
/usr/lib/gcc/i686-linux-gnu/4.4.5/../../../../lib/crt1.o: In function `\_start': (.text+0x18): undefined reference to `main'

collect2: ld returned 1 exit status

* Lỗi trên xuất hiện vì tập tin cong.c không có hàm main() nên không thể tạo ra chương trình thực thi được.
* Tập tin cong.c chỉ có thể dịch ra thành tập tin mã đối tượng mà thôi (có phần mở rộng là .o)
* Dùng tùy chọn -c để dịch tập tin cong.c thành tập tin mã đối tượng

## gcc -c cong.c

* Đánh lệnh **ls** để xem nội dung thư mục Bai1, bạn sẽ thấy tập tin **cong.o** mới được tạo ra



# Bài 2: Viết chương trình cong2so.c có hàm main() gọi hàm int cong(int a, int b) đã được tạo ra ở bài 1 phía trên.

* Tạo thư mục **Bai2** (QUAN TRỌNG)
* Lưu nội dung sau vào tập tin **cong2so.c** đặt trong thư mục Bai2 (QUAN TRỌNG)

#include<stdio.h> int main()

{

int a=5; int b=10; int ketqua;

ketqua = cong(a,b); printf("%d+%d=%d\n",a,b,ketqua);

}

* Đánh lệnh sau để biên dịch chương trình cong2so.c trên thành chương trình thực thi có tên là

## cong2so

* + **gcc -o cong2so cong2so.c**
* Bạn sẽ gặp thông báo lỗi sau:

In function `main':

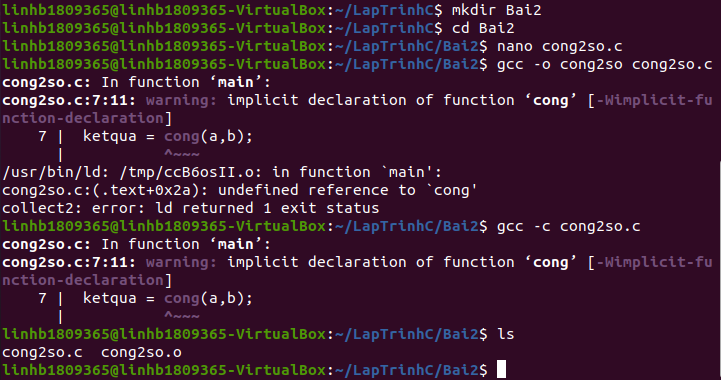
cong2so.c:(.text+0x29): undefined reference to `cong' collect2: ld returned 1 exit status

Vì bạn có gọi đến hàm cong nằm trong tập tin cong.c nhưng khi biên dịch bạn không đưa thêm tập tin cong.c vào dòng lệnh biên dịch

* Hãy đánh dòng lệnh sau để biên dịch cong2so.c thành tập tin mã đối tượng cong2so.o

## gcc -c cong2so.c

* Đánh lệnh **ls** để xem nội dung của thư mục Bai2 có chứa tập tin cong2so.o hay không

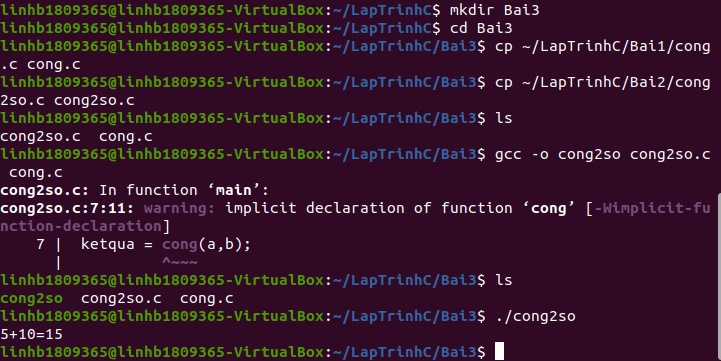


# Bài 3:Tạo tập tin thực thi từ các tập tin nguồn .c

* Hãy tạo thư mục **Bai3**
* Copy tập tin **cong.c** từ thư mục Bai1 vào thư mục Bai3
* Copy tập tin **cong2so.c** từ thư mục Bai2 vào thư mục Bai3
* Mở terminal
* Chuyển vào thư mục Bai3 (QUAN TRỌNG)
* Đánh lệnh **ls** để xem nội dung thư mục Bai3 có chứa 2 tập tin cong.c và cong2so.c hay không.
* Dùng lệnh sau để tạo chương trình thực thi có tên cong2so từ 2 tập tin nguồn cong2so.c và cong.c

## gcc -o cong2so cong2so.c cong.c

* Đánh lệnh **ls** để xem sự hiện diện của tập tin cong2so vừa mới tạo ra.
* Đánh lệnh sau để thực hiện chương trình cong2so: **./cong2so**

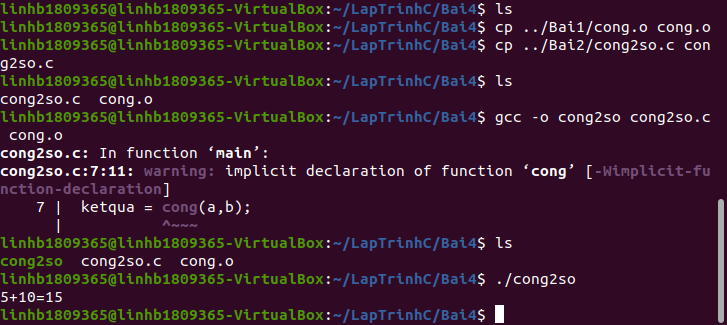


# Bài 4: Tạo tập tin thực thi từ các tập tin nguồn .c và các tập tin mã đối tượng

* Hãy tạo thư mục **Bai4**
* Copy tập tin **cong.c** từ thư mục Bai1 vào thư mục Bai4
* Copy tập tin **cong2so.c** từ thư mục Bai2 vào thư mục Bai4
* Mở terminal
* Chuyển vào thư mục Bai4 (QUAN TRỌNG)
* Đánh lệnh **ls** để xem nội dung thư mục Bai4 có chứa 2 tập tin cong.o và cong2so.c hay không.
* Dùng lệnh sau để tạo chương trình thực thi có tên cong2so từ 2 tập tin nguồn cong2so.c và cong.c

## gcc -o cong2so cong2so.c cong.c

* Đánh lệnh **ls** để xem sự hiện diện của tập tin cong2so vừa mới tạo ra.
* Đánh lệnh sau để thực hiện chương trình cong2so: **./cong2so**



# Bài 5: Tạo tập tin thư viện hàm

* Tạo thư mục **Bai5** trong thư mục LapTrinhC
* Mở slide bài giảng chủ đề Lập trình C/Linux, Di chuyển đến slide số 12.
* Tạo 4 tập tin với tên và nội dung như slides số 12. Lưu cả 4 tập tin trong thư mục Bai5:

## mylib.h

* + **hello.c**
  + **bonjour.c**
  + **helloworld.c**
* Mở terminal; Chuyển vào thư mục Bai5 (QUAN TRỌNG)
* Dùng lệnh sau để biên dịch các tập tin .c ra tập tin mã đối tượng

## gcc -c \*.c

* Đánh lệnh **ls** để xem các tập tin .o nào vừa mới tạo ra.
* Đánh lệnh sau để tạo tập tin hàm thư viện có tên là **libmylib.a**

## ar crv libmylib.a hello.o bonjour.o

* Đánh lệnh **ls** để xem có tập tin libmylib.a trong thư mục Bai5 không.
* Đánh lệnh sau để tạo tập tin thư viện **helloworld** từ tập tin hellworld.o và tập tin thư viện libmylib.a

## gcc -o helloworld helloworld.o libmylib.a

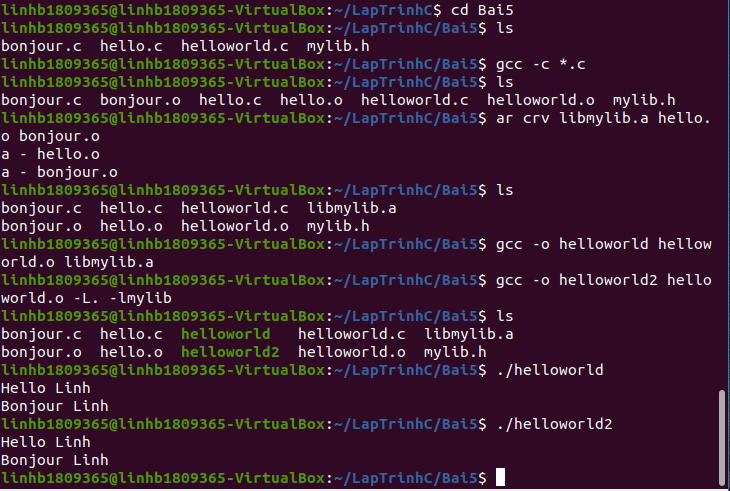
* Có thể sử dụng biểu hiện của tập tin hàm thư viện libmylib.a khi biên dịch chương trình như lệnh sau:

## gcc -o helloworld2 helloworld.o -L. -lmylib

* Đánh lệnh **ls** để xem những tập tin nào vừa được tạo ra trong thư mục Bai5
* Thực thi chương trình helloworld hoặc hellowolrd2

## ./helloworld

* + **./helloworld2**



**Bài 6: Tạo tập tin makefile**

* Tạo thư mục **Bai6** trong thư mục LapTrinhC
* Copy tất cả các tập tin **.c** và **.h** trong thư mục Bai5 sang thư mục Bai6
* Trong thư mục Bai6, tạo tập tin có tên là **makefile** với nội dung như sau ( Lưu ý, khi copy về phải thay thế các khoảng trắng phía trước lệnh gcc bằng 1 phím TAB):

helloworld: helloworld.o mylib

gcc -o helloworld helloworld.o libmylib.a helloworld.o: helloworld.c

gcc -c helloworld.c mylib: hello.o bonjour.o

ar crv libmylib.a hello.o bonjour.o hello.o: hello.c

gcc -c hello.c bonjour.o: bonjour.c

gcc -c bonjour.c

* Chuyển vào thư mục Bai6
* Đánh lệnh **ls** để xem nội dung thư mục Bai6
* Đánh lệnh sau để tạo tập tin helloworld bằng tiện ích make

## make

* Đánh lệnh **ls** để những tập tin nào vừa được tạo ra bởi make xem nội dung thư mục Bai6
* Thực thi chương trình helloworld

## ./helloworld

## 

**Bài 7: Thêm một hàm mới vào thư viện**

* Tạo thư mục **Bai7** trong thư mục LapTrinhC
* Copy tất cả các tập tin **.c**, **.h** và tập tin **makefile** trong thư mục Bai6 sang thư mục Bai7
* Tạo tập tin có tên là **chao.c** trong thư mục Bai7, với nội dung như sau:

char\* chao()

{

return "Chao";

}

* Sửa nội dung các tập tin trong thư mục Bai7 như sau:

+ Thêm dòng sau vào cuối tập tin **mylib.h** để khai báo hàm mới cho thư viện

## char\* chao();

+ Thêm dòng sau vào trước lệnh return của tập tin helloworld.c

## printf("%s Hung\n",chao());

+ Cập nhật lại nội dung tập tin makefile như sau (Lưu ý, khi copy về phải thay thế các khoảng trắng phía trước lệnh gcc bằng 1 phím TAB):

helloworld: helloworld.o mylib

gcc -o helloworld helloworld.o libmylib.a helloworld.o: helloworld.c

gcc -c helloworld.c

mylib: hello.o bonjour.o chao.o

ar crv libmylib.a hello.o bonjour.o chao.o hello.o: hello.c

gcc -c hello.c bonjour.o: bonjour.c

gcc -c bonjour.c chao.o: chao.c

gcc -c chao.c

* Mở một Terminal; Chuyển vào thư mục Bai7
* Đánh lệnh sau để tạo lại tập tin helloworld

## make

* Đánh lệnh sau để thực thi chương trình helloworld

## ./helloworld

