**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO BÀI TẬP NMHĐH**

**HK2, 2021-2022**

**Lab 2**

**Nhóm:** 05 **Tổ:** 01

**Thành Viên Nhóm:**

1: Nguyễn Trần Quang Huy (MSSV: 52000668)

**Mục lục**

**A. PHẦN THỰC HÀNH**

**Ví Dụ 1:** Biên dịch chương trình đầu tiên

**A: Code Chương Trình**

// hello.c

#include <stdio.h>

int main() {

printf("Hello World\n");

return 0;

}

Code

**B: Kết Quả Demo**

Text

Description automatically generated

* **Ví Dụ 2: Truyền đối số từ lệnh gọi**

**2.1: Code chương trình**

// para.c

#include <stdio.h>

int main(int argc, char \*\*argv) {

printf("Number of arguments: %i \n", argc);

int i = 0;

for (i = 0; i < argc; i++)

printf("Argument[%i] %s\t", i, argv[i]);

printf("\n");

return 0;

}

**2.2 Kết quả Demo**

Text

Description automatically generated

* **Ví dụ 3: Liên kết tĩnh**

**3.1 Code chương trình**

// hello1.c

#include <stdio.h>

void hello\_1(int i)

{

printf("Hello, parameter 1 = %d\n", i);

}

// hello2.c

#include <stdio.h>

void hello\_2(int i)

{

    printf("Hello, parameter 2 = %d\n", i);

}

// main.c

#include <stdio.h>

#include <stdlib.h>

int main(int argc, char \*\*argv)

{

    int i = atoi(argv[1]);

    int k = atoi(argv[2]);

    hello\_1(i);

    hello\_2(k);

}

**3.2 Kết quả demo**

**Text

Description automatically generated**

**B. PHẦN BÀI TẬP**

**BÀI 1:** Viết chương trình sau cho khi truyền đối số n vào thì xuất ra tổng S = 1 + 2 + …. + n

1. Báo lỗi nếu lời gọi có đối số không phải là một số nguyên dương.
2. Báo lỗi nếu có nhiều hơn 2 đối số (là main.out và n).

**A: Code Chương Trình:**

// main.c

#include <stdio.h>

int main(int argc, char \*\*argv) {

if (argc > 2) { //nếu số lượng đối số lớn hơn hai là sai

printf("Co qua nhieu doi so");

}

else if (atoi(argv[1]) <= 0) { //đối số truyền vào không phải là số dương

printf("Doi so khong phai la so nguyen duong");

}

else {

//tính tổng từ 0 đến n

int sum = 0;

for (int i = 0; i < atoi(argv[1]); i++) {

sum += (i + 1);

}

printf("s = %d", sum);

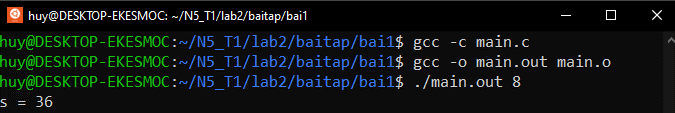
}

printf("\n");

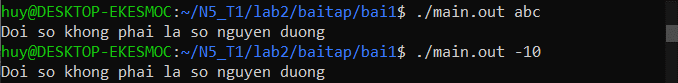
return 0;

}

**B: Kết Quả Demo:**

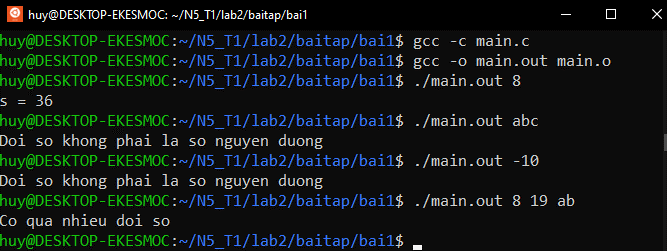


1. Báo lỗi nếu lời gọi có đối số không phải là một số nguyên dương.



1. Báo lỗi nếu có nhiều hơn 2 đối số (là main.out và n).





**BÀI 2:** Viết chương trình truyền vào một số nguyên, và in ra dãy các ước số của số nguyên này.

1. Báo lỗi nếu đối số không phải là số nguyên, hoặc thừa đối số.
2. Phân tích số nguyên đã truyền vào thành thừa số nguyên tố. (bài tập nâng cao).

**A: Code Chương Trình:**

// main.c

#include <stdio.h>

#include <stdlib.h>

int main(int argc, char \*\*argv) {

int n = atoi(argv[1]);

if (argc > 2) { //nếu số lượng đối số lớn hơn hai là sai

printf("Co qua nhieu doi so");

}

else if (n <= 0) { //đối số không phải là số nguyên dương sai

printf("Doi so khong phai la so nguyen duong");

}

else { //liệt kê qua các ước số của n nếu n chia i dư 0

printf("Cac uoc so cua %d la ", n);

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

if (n % i == 0) {

printf("%d, ", i);

}

}

int prime = 2; //ta sẽ cho số nguyên tố nhỏ nhất là 2 và lấy n chia dần

int count = 0; //cứ sau mỗi lần sẽ tăng prime lên 1 nhưng sẽ không có vụ là sẽ chia hết cho 4 vì đã chia cho 2, 2 lần rồi tương tự vậy cho các số khác không phải là số nguyên tố khi tăng prime lên 1 sau cùng chỉ cần đếm số lần chia hết cho prime sau đó in theo mẫu prime^count là được

printf("\n%d = ", n);

while (1) {

if (n % prime == 0) {

n /= prime;

count++;

continue;

}

else if (n == 1) {

if (count == 1) {

printf("%d", prime);

}

else {

printf("%d^%d", prime, count);

}

break;

}

else {

if (count == 1) {

printf("%d\*", prime);

}

else if (count > 1) {

printf("%d^%d\*", prime, count);

}

}

count = 0;

prime++;

}

}

printf("\n");

return 0;

}

**B: Kết Quả Demo:**

Text

Description automatically generated

1. Báo lỗi nếu đối số không phải là số nguyên, hoặc thừa đối số.

Text

Description automatically generated

1. Phân tích số nguyên đã truyền vào thành thừa số nguyên tố. (bài tập nâng cao).

Text

Description automatically generated

**BÀI 3:** Viết chương trình truyền vào một danh sách số nguyên, và in ra dãy số này theo thứ tự  
tăng dần.

1. Bỏ qua các đối số không phải là số nguyên.
2. Hãy áp dụng các thuật toán sắp xếp đã học. (bài tập về nhà).

**A: Code Chương Trình:**

// main.c

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

int main(int argc, char \*\* argv) {

int size = 0; //đầu tiên là đếm số lượng số nguyên kể cả số 0

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

//hàm strcmp là hàm so sánh 2 chuỗi trả về 0 nếu 2 chuỗi bằng nhau

if (atoi(argv[i]) != 0 || strcmp(argv[i], "0") == 0) {

size++;

}

}

int a[size]; //tạo 1 array có kích thước là size đã đếm

int index = 0; //sau đó thêm từ phần tử hợp lệ vào array

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

if (atoi(argv[i]) != 0 || strcmp(argv[i], "0") == 0) {

a[index++] = atoi(argv[i]);

}

}

for (int i = 0; i < size - 1; i++) { //áp dụng thuật toán bubble sort chưa cải tiến để sắp xếp mảng số nguyên này

for (int j = i + 1; j < size; j++) {

if (a[i] > a[j]) {

int temp = a[i];

a[i] = a[j];

a[j] = temp;

}

}

}

printf("Day tang la "); //công việc cuối cùng là in ra thôi

for (int i = 0; i < size; i++) {

printf("%d ", a[i]);

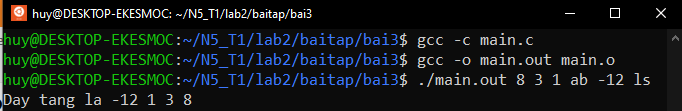
}

printf("\n");

return 0;

}

**B: Kết Quả Demo:**



**BÀI 4:** Viết các tập tin add.c, sub.c lần lượt chứa 2 hàm số int add(int a, int b) và int sub(int a, int b).

1. Xây dựng một thư viện liên kết tĩnh từ 2 tập tin add.c và sub.c.
2. Xây dựng một thư viện liên kết động từ 2 tập tin add.c và sub.c.

**A: Code Chương Trình:**

// add.c //tạo file add.c

#include <stdio.h>

#include <stdlib.h>

int add(int a, int b) {

return a + b;

}

// sub.c //tạo file sub.c

#include <stdio.h>

#include <stdlib.h>

int sub(int a, int b) {

return a - b;

}

// main.c

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

int main(int argc, char \*\* argv) {

if (argc != 4) { //số lượng tham số truyền vào nhiều quá

printf("Doi so truyen khong dung.");

} else {

int a = atoi(argv[1]); // convert sang int

int b = atoi(argv[2]); // convert sang int

char \* oper = argv[3];

if (strcmp(oper, "+") == 0) {//hàm strcmp so sánh 2 chuỗi tương tự câu3

printf("Ket qua: %d", add(a, b));

} else if (strcmp(oper, "-") == 0) {

printf("Ket qua: %d", sub(a, b));

} else {

printf("Doi so truyen khong dung.");

}

}

printf("\n");

return 0;

}

**B: Kết Quả Demo:**

1. Xây dựng một thư viện liên kết tĩnh từ 2 tập tin add.c và sub.c.

Text

Description automatically generated

1. Xây dựng một thư viện liên kết động từ 2 tập tin add.c và sub.c.

Text

Description automatically generated

**KẾT LUẬN**

**Sau khi học và hoàn thành phần lab 2 nhóm thu được kết sau:**

* Cách chạy một chương trình .c trên linux.
* Truyền đối số từ lệnh gọi chứ không phải qua scanf.
* Các build thư viện liên kết tĩnh và liên kết động.

**KẾT QUẢ LÀM VIỆC CÁC THÀNH VIÊN TRONG NHÓM**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HỌ TÊN SV | MÃ SV | VAI TRÒ | CÔNG VIỆC ĐƯỢC PHÂN CÔNG | HT |
| Nguyễn Trần Quang Huy | 52000668 | TN | Làm cả phần thực hành và bài tập | 100% |
|  |  |  |  |  |

**CHÚ THÍCH:**

TN: TRƯỞNG NHÓM

TV: THÀNH VIÊN

HT: MỨC ĐỘ HOÀN THÀNH CÔNG VIỆC ĐƯỢC GIAO