



ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

LẬP TRÌNH C CƠ BẢN

Chương 1. Khởi động & nhập xuất dữ liệu với tệp

Soạn bởi: TS. Nguyễn Bá Ngọc
(Dựa trên bài giảng sử dụng chung)

Giới thiệu về môn học

- ❖ Lập trình theo các chủ đề về cấu trúc dữ liệu và giải thuật ở mức độ cơ bản và ứng dụng vào giải quyết một số bài toán thực tế.
- ❖ Ngôn ngữ lập trình: C

Giới thiệu về môn học

- ❖ Môi trường khuyến nghị:
 - Hệ điều hành Linux: Ubuntu
 - Trình biên dịch: GCC (GNU)
 - Ngoài ra còn có: Clang
 - Trình soạn thảo mã nguồn: Sublime Text
 - Ngoài ra còn có: Visual studio code, Emacs v.v.

Ví dụ biên dịch bằng gcc

Câu lệnh biên dịch: `gcc -Wall -o hello hello.c`

Chạy chương trình (Linux): `./hello`

- ❖ Các tham số:
 - Wall : bật tất cả các cảnh báo
 - o: tạo tập tin chương trình
 - g: thêm thông tin gỡ rối

Chủ đề

- ❖ Ôn tập NNLT C:
 - Mảng,
 - Xâu/Chuỗi ký tự,
 - Con trỏ.
- ❖ Sử dụng đối số dòng lệnh
- ❖ Thao tác với tập tin văn bản
 - Đọc/ghi theo từng ký tự
 - Đọc/ghi theo từng dòng
 - Đọc/ghi theo định dạng
- ❖ Các bài tập lập trình

Chủ đề

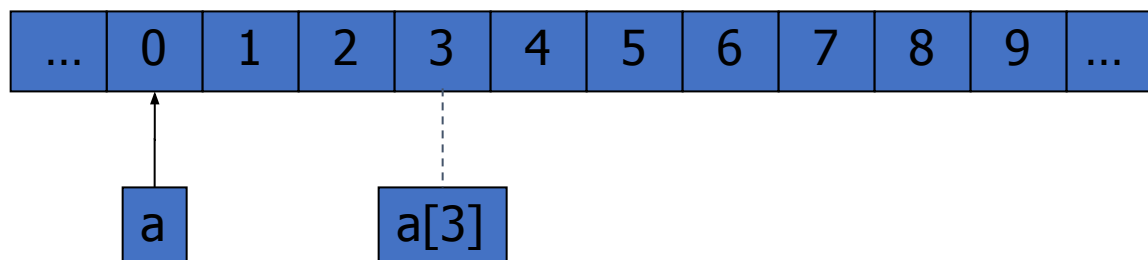
- ❖ Ôn tập NNLT C
 - Mảng,
 - Xâu/Chuỗi ký tự,
 - Con trỏ.
- ❖ Sử dụng đối số dòng lệnh
- ❖ Thao tác với tập tin văn bản
 - Đọc/ghi theo từng ký tự
 - Đọc/ghi theo từng dòng
 - Đọc/ghi theo định dạng
- ❖ Các bài tập lập trình

Mảng

- ❖ Khối dữ liệu liên tục trong bộ nhớ
- ❖ Các ví dụ về sử dụng mảng
 - Danh sách sinh viên trong lớp học
 - Dãy số nhập từ người dùng

Biểu diễn mảng trong bộ nhớ

- ❖ Khối dữ liệu liên tục trong bộ nhớ
- ❖ Mảng chứa địa chỉ của ô nhớ đầu tiên
- ❖ Ví dụ khai báo mảng:
double a[10];
- ❖ Các phần tử của một mảng n phần tử được đánh chỉ số từ 0 đến n - 1.



Ví dụ 1.1. hiển thị mảng theo chiều ngược

```
vd1-1.c x
1  #include <stdio.h>
2
3  #define N 10
4
5  int main(void){
6      double a[N];
7      printf("Mời nhập %d số:\n", N);
8      for(int i=0; i< N; i++) {
9          scanf("%lf", &a[i]);
10     }
11     printf("Các số theo thứ tự ngược lại:\n");
12     for(int i = N - 1; i>=0; --i) {
13         printf("%g\n", a[i]);
14     }
15     return 0;
16 }
17
```

Bài tập 1.1. Giá trị gần giá trị trung bình nhất

- ❖ Viết tiếp chương trình ví dụ 1.1 hiển thị tất cả các phần tử có giá trị gần nhất so với giá trị trung bình của tất cả các phần tử mảng.

Nhập: 1 2 3 4 5 6 7 8 9 10

trung bình = 5.5

Xuất: 5 (a[4]), 6 (a[5])

Hướng dẫn: Giải thuật (mã giả)

Tính **avg** = trung bình các phần tử của mảng a.

Khởi tạo: $\text{min} = \text{fabs}(a[0] - \text{avg});$

Duyệt mảng để tìm khác biệt tối thiểu so với **avg** (tìm **min**).

Duyệt mảng và xuất ra các phần tử có khác biệt so với **avg** bằng khác biệt tối thiểu.

Bài tập 1.2. Mảng ký tự

- ❖ Viết chương trình nhập vào các ký tự cho tới khi gặp dấu '\n'.
- ❖ Chương trình cần đếm số lần xuất hiện của các chữ cái được nhập vào và xuất ra số lần xuất hiện của các chữ cái đó. Bỏ qua tất cả các ký tự khác không phải là chữ cái.

Ví dụ kết quả hiển thị với dữ liệu đầu vào: “hello, world!\n”

Chữ 'd' xuất hiện 1 lần.
Chữ 'e' xuất hiện 1 lần.
Chữ 'h' xuất hiện 1 lần.
Chữ 'l' xuất hiện 3 lần.
Chữ 'o' xuất hiện 2 lần.
Chữ 'r' xuất hiện 1 lần.
Chữ 'w' xuất hiện 1 lần.

Bài tập1.3. Hàm có mảng là đối số

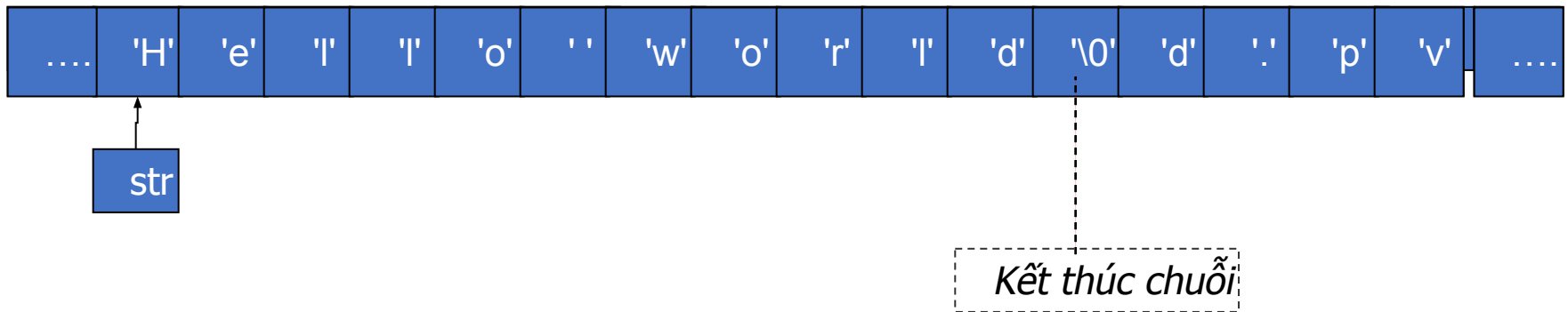
- Viết một hàm nhận các tham số đầu vào là hai mảng số nguyên có cùng số lượng phần tử. Hàm trả về 1 nếu hai mảng có nội dung trùng nhau và 0 nếu ngược lại (kiểm tra tuần tự theo từng phần tử).
- Viết chương trình minh họa hoạt động của hàm thu được.

Ví dụ:

1. Nếu truyền vào {1, 2, 3} và {1, 2, 3} thì hàm trả về 1.
2. {1, 2, 3} và {1, 3, 3} thì hàm trả về 0.

Xâu ký tự trong C

- Được biểu diễn như mảng ký tự với quy ước sử dụng ký tự có mã số = 0 ('\0') làm dấu hiệu kết thúc.



Chuỗi ký tự

- Để lưu một chuỗi ký tự có N ký tự chúng ta cần sử dụng mảng có độ dài tối thiểu là $N + 1$.
- Các khai báo chuỗi sau là tương đương:

```
char str[] = "Hanoi";
```

```
char str[] = {'H', 'a', 'n', 'o', 'i', '\0'};
```

```
char str[6] = "Hanoi";
```

```
char str[6] = {'H', 'a', 'n', 'o', 'i', '\0'};
```

```
char str[6] = {'H', 'a', 'n', 'o', 'i'};
```

```
str[5] = ?? ('\0' tương đương 0)
```

Một số hàm nhập ký tự và chuỗi ký tự

stdio.h

- getchar - Trả về một ký tự được nhập vào
- scanf - Đa dụng, %c - nhập ký tự, %s nhập chuỗi không chứa khoảng trắng, %[^\n]*c - Nhập vào chuỗi có chứa khoảng trắng cho tới khi gặp \n và bỏ qua \n (tương tự gets).
- fgets - Nhập một dòng văn bản.

Một số hàm xử lý chuỗi ký tự

string.h

- strlen - Trả về số ký tự của chuỗi s.
- strcmp - So sánh 2 chuỗi theo thứ tự chữ cái
- strcpy - Sao chép nội dung s2 vào s1
- strcat - Nối s2 vào sau s1

Bài tập 1.4. Đọc và xử lý chuỗi ký tự

- Viết hàm :
 - Có tham số là một chuỗi ký tự và hai ký tự
 - Duyệt chuỗi và thay thế tất cả các ký tự thứ nhất trong chuỗi bằng ký tự thứ hai.
- Viết chương trình để kiểm tra hàm nói trên:
 - Đọc một chuỗi có độ dài tối đa là 30 ký tự, không chứa ký tự trắng và hai ký tự, sau đó gọi hàm với các đối số trên và in ra kết quả.
- Ví dụ
 - Đầu vào: “papa”, ‘p’, ‘m’
 - Kết quả: “mama”

Bài tập 1.5. Tách từ

- Viết chương trình đọc một chuỗi ký tự biểu diễn một câu từ người dùng. Sau đó chương trình hiển thị mỗi từ trong câu trên một dòng theo thứ tự ngược lại. Một từ là một dãy các ký tự liên tiếp không chứa ký tự trắng và không chứa dấu chấm câu (.).
- Ví dụ:
 - Đầu vào: The house next door is very old.
 - Kết quả:
 - old
 - very
 - is
 - door
 - next
 - house
 - The

Kiểu con trỏ và biến con trỏ

- ❖ Kiểu con trỏ là một kiểu được suy diễn từ một kiểu khác (kiểu cơ sở của kiểu con trỏ). Kiểu con trỏ mô tả đối tượng có giá trị là tham chiếu/địa chỉ bộ nhớ của một đối tượng thuộc kiểu cơ sở của nó. Kiểu con trỏ được suy diễn từ một kiểu T thường được gọi là kiểu con trỏ tới T.
 - Khái niệm này có thể được áp dụng đệ quy.
- ❖ Chúng ta gọi tắt biến có kiểu con trỏ là con trỏ
- ❖ Ví dụ:

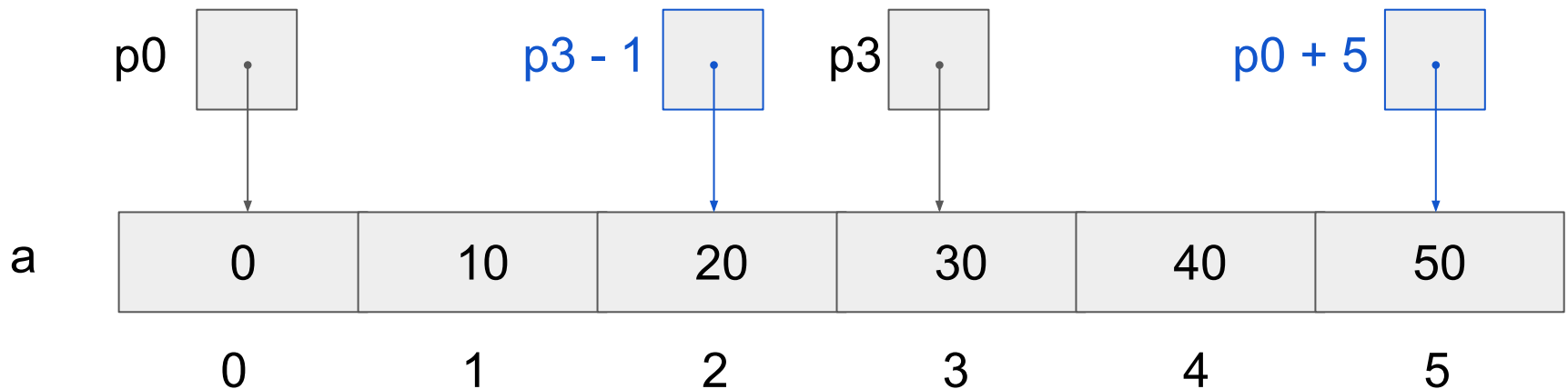
```
int x;  
int *p = &x;
```

 - `int *` là kiểu con trỏ tới `int`
 - `p` là biến có kiểu `int *`/biến có kiểu con trỏ tới `int`/con trỏ tới biến kiểu `int`.
 - `p` là con trỏ tới `x`.

Ví dụ con trỏ và đại số con trỏ

```
int a[6] = {0, 10, 20, 30, 40, 50};  
int *p0 = &a[0], *p3 = &a[3];
```

```
p0 - p3 == -3  
*(p3 - 1) == 20  
*(p0 + 5) == 50
```



Bài tập 1.6. Hàm trả về 2 giá trị

- ❖ Viết hàm nhận đối số là một số thực (double) và trả về phần nguyên và phần thập phân của số đó.
- ❖ Viết chương trình minh họa hàm trên, với số thực được nhập từ người dùng.

1. `split(3.14, &a, &b);`

2. `struct x = split(3.14);`

`x.a, x.b`

`=> a = 3; b = 0.14;`

Bài tập 1.7. Sinh câu văn tự động

- Viết chương trình có khả năng sinh ra các câu tự động sử dụng kỹ thuật lựa chọn dựa trên số ngẫu nhiên. Chương trình dùng bốn mảng xâu ký tự để lưu trữ **các mạo từ (article), danh từ (noun), động từ (verb), giới từ (preposition)**. Câu được tạo ra bằng cách lựa chọn ngẫu nhiên các phần tử trong các mảng trên và ghép lại theo thứ tự: **mạo từ, danh từ, động từ, mạo từ và danh từ**. Câu sinh ra cần bắt đầu với ký tự in hoa và kết thúc với dấu chấm. Chương trình cần sinh ra tối thiểu 10 câu.
- Ví dụ về các phần tử mảng
 - mạo từ: "the", "a", "one", "some" and "any";
 - danh từ: "boy", "girl", "dog", "town" and "car";
 - động từ: verbs "drove", "jumped", "ran", "walked" and "skipped";
 - giới từ: "to", "from", "over", "under" and "on".

Chủ đề

- ❖ Ôn tập NNLT C
 - Mảng,
 - Xâu/Chuỗi ký tự,
 - Con trỏ.
- ❖ Sử dụng đối số dòng lệnh
- ❖ Thao tác với tập tin văn bản
 - Đọc/ghi theo từng ký tự
 - Đọc/ghi theo từng dòng
 - Đọc/ghi theo định dạng
- ❖ Các bài tập lập trình

SỬ DỤNG ĐỐI SỐ DÒNG LỆNH

- ❖ Có thể truyền các tham số cho hàm **main** từ dòng lệnh được sử dụng để thực thi chương trình.
- ❖ Nguyên mẫu hàm **main** nhận các tham số được truyền vào từ dòng lệnh (nguyên mẫu đầy đủ):

```
int main(int argc, char *argv[])
```

Số lượng tham số

Mảng các con trỏ tới các tham số được biểu diễn như các chuỗi ký tự.

*Hàm **main** là một hàm đặc biệt, một chương trình C phải có đúng một hàm main và đó cũng là điểm bắt đầu thực hiện chương trình.*

Ví dụ 1.2. Truyền tham số cho hàm main

```
vd1-2.c x
1 #include <stdio.h>
2 int main(int argc, char *argv[]) {
3     for (int i = 0; i < argc; ++i) {
4         printf("argv[%d] = %s\n", i, argv[i]);
5     }
6     return 0;
7 }
```

gcc -o p vd1-2.c

./p 1 2.2 s3 "Tham so chua dau cach"

argv[0] = ./p

argv[1] = 1

argv[2] = 2.2

argv[3] = s3

argv[4] = Tham so chua dau cach

Bài tập 1.8. Chu vi HCN

- Viết chương trình nhận hai số thực làm đối số dòng lệnh, đại diện cho chiều dài và chiều rộng của một hình chữ nhật.
- Chương trình tính toán và in ra diện tích và chu vi của hình chữ nhật.
- Ví dụ đặt tên chương trình là hcn:
 - ./hcn 10 20 - cho kết quả là 200;
 - ./hcn 15 6 - cho kết quả là 90.

Bài tập 1.9. Đảo ngược trật tự đối số

- Viết chương trình hiển thị các đối số dòng lệnh do người dùng nhập vào theo thứ tự ngược lại.
- Ví dụ đặt tên chương trình là Inverse
 - ./inverse I love HUST - cho kết quả: HUST love I

Bài tập 1.10. Lũy thừa

- ❖ Viết chương trình tính lũy thừa x của cơ số tự nhiên e (e^x). Người dùng có thể chạy chương trình theo 2 cú pháp tương tự như trong ví dụ.
- ❖ Ví dụ đặt tên chương trình là `ex`
 - `./ex 3` - chương trình có thể cho kết quả 20.085536923187668
 - `./ex 3 6` - chương trình cho kết quả 20.085537

x

số chữ số phần
thập phân

Bài tập 1.11. Phương trình bậc 2

- Viết chương trình có tên pb2 nhận đối số dòng lệnh là các hệ số của một phương trình bậc 2 ($ax^2 + bx + c = 0$) và giải phương trình. Sau đó in ra màn hình các nghiệm.
- Ví dụ đặt tên chương trình là pb2
 - ./pb2 1 2 1 cho kết quả: $x_1 = x_2 = -1$



25 YEARS ANNIVERSARY
SOICT

VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG
SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

**Thank you
for your
attentions!**



soict.hust.edu.vn/



fb.com/groups/soict

