

## A. Câu hỏi

1. Cho đoạn lệnh sau.

```
char c = 'A', d = 'B';  
char* p1 = &c;  
char* p2 = &d;  
char* p3;
```

Giả sử địa chỉ của c là 0x1234, địa chỉ của d là 0x5678. Hỏi output của đoạn lệnh sau là gì?

```
p3 = &d;  
cout << "*p3 = " << *p3 << ", p3 = " << p3 << endl;  
// in ra *p3 = B , p3 = 0x5678  
p3 = p1;  
cout << "*p3 = " << *p3 << ", p3 = " << p3 << endl;  
// in ra *p3 = A , p3 = 0x1234  
*p1 = *p2;  
cout << "*p1 = " << *p1 << ", p1 = " << p1 << endl;  
// in ra *p1 = B , p1 = 0x1234
```

2. Cho các lệnh sau.

```
int *p;  
int i;  
int k;  
i = 4;  
k = i;  
p = &i;
```

(Những) lệnh nào trong số các lệnh dưới đây sẽ gán cho i giá trị 5?

```
k = 5;  
*k = 5;  
p = 5;  
*p = 5; câu lệnh này có thể gán cho i giá trị 5;
```

3. Giải thích lỗi sau:

```
char c = 'C';  
double *p = &c;
```

Lỗi biên dịch bởi biến c đang có kiểu dữ liệu char nhưng con trỏ khai báo để trỏ tới biến c lại có kiểu dữ liệu là double.

## B. Bài tập

1. Tất cả các câu trong bài này đều phải dùng con trỏ để xử lý mảng/xâu, KHÔNG dùng thư viện C++string, hay cstring, hãy tự viết lấy các hàm bạn cần sử dụng, chẳng hạn strlen, strcpy...  
Viết các hàm sau đây, demo việc sử dụng chúng trong hàm main() của một chương trình.
  - a. **Đảo xâu.** Viết một hàm `reverse(char a[])` với nhiệm vụ đảo ngược thứ tự của a).
  - b. **Xóa kí tự.** Viết hàm `delete_char(char a[], char c)` với nhiệm vụ xóa hết các kí tự c trong xâu a.
  - c. **Độn phải.** Viết hàm `pad_right(char a[], int n)` với nhiệm vụ độn thêm các kí tự space vào cuối xâu a để a có độ dài bằng n. Nếu a đã dài quá n thì không phải làm gì cả.
  - d. **Độn trái.** Viết hàm `pad_left(char a[], int n)` với nhiệm vụ độn thêm các kí tự space vào đầu xâu a để a có độ dài bằng n. Nếu a đã dài quá n thì không phải làm gì cả.
  - e. **Cắt xâu.** Viết hàm `truncate(char a[], int n)` với nhiệm vụ cắt bớt đoạn cuối của xâu a để a có độ dài n nếu như a đang dài quá n kí tự.
  - f. **Đối gương.** Viết hàm `bool is_palindrome(char a[])` với nhiệm vụ kiểm tra xem xâu a có đối xứng hay không. Trả về true nếu đối xứng, false nếu không.
  - g. **Lọc trái.** Viết hàm `trim_left(char a[])` với nhiệm vụ xóa các kí tự trắng đứng ở đầu xâu a, nếu có.
  - h. **Lọc phải.** Viết hàm `trim_right(char a[])` với nhiệm vụ xóa các kí tự trắng đứng ở cuối xâu a, nếu có.

**Đã làm và nộp trên github**

## D. Câu hỏi trắc nghiệm

1. Câu lệnh nào sau đây thực hiện gán giá trị của n cho con trỏ \*p  
`int n = 5; int *p;`
  - i. `p = n;`
  - ii. **`p = &n;`**
  - iii. `*p = n;`
  - iv. `*p = &n;`
2. Câu lệnh nào sau đây thực hiện chỉ con trỏ \*p1 vào vị trí con trỏ \*p2 đang chỉ
  - i. `*p1 = p2;`
  - ii. `p1 = *p2;`
  - iii. **`p1 = p2;`**
  - iv. `*p1 = *p2`
3. Đối với phép toán con trỏ, phát biểu nào sau đây có ý nghĩa lập trình
  - i. Phép nhân trên con trỏ kiểu int
  - ii. **Phép cộng trừ trên con trỏ kiểu int**
  - iii. **Phép gán với con trỏ kiểu int**

iv. Tất cả các phép toán với con trỏ kiểu int

4. Phát biểu nào sau đây chính xác

i. Phép cộng trừ với các con trỏ ở mảng khác nhau

ii. **Phép cộng trừ với các con trỏ ở cùng mảng**

iii. **Tất cả các phép toán với các con trỏ ở cùng mảng (trong giới hạn mảng)**

iv. Tất cả các phát biểu trên

5. Phát biểu nào sau đây đúng với con trỏ tới hàm

i. Có thể được truyền vào trong trong

ii. Có thể được trả về bởi hàm

iii. Có thể được lưu trong mảng

iv. **Tất cả các phát biểu trên**

6. Khai báo hàm nào sau đây không đúng

i. void bubble(int [], const int, bool (\*)(int, int));

ii. **void bubble(int [], const int, bool \* (int, int));**

iii. **void bubble(int [], const int, (bool \*) (int, int));**

iv. **void bubble(int [], const int, bool (\* (int, int)));**

7. Kết quả đoạn mã sau là gì:

```
int a[6] = {1, 2, 3};
int *b; b = &a[0];
for (int i=0; i<3; i++)
    switch (i%3) {
        case 0: *(b+i) = a[i]; break;
        case 1: *(b++) = a[i]; break;
        case 2: *(++b) = a[i]++; break;
    }
for (int i=0; i<3; i++) cout << a[i] << " ";
```

i. 0 2 3

ii. 2 2 0

iii. **2 2 3**

iv. 3 0 3