

Bài tập môn Lập trình Hướng đối tượng với C++

Chương 1 & 2: Lập trình cơ bản

Bài tập 1: Viết chương trình nhập vào từ bàn phím một số n và in ra ma trận theo qui luật sau:

- với $n = 1$ in ra ma trận gồm 1 số

10

- với $n = 2$ in ra ma trận gồm 3 số :

10	
10	20

- với $n = 3$ in ra ma trận gồm 5 số:

10		
10	20	
10	20	30

Bài tập 2: Viết chương trình nhập vào 1 số từ bàn phím và in ra ma trận theo qui luật sau:

- $n = 1$ in ra ma trận gồm 1 số:

1

- $n = 2$ in ra ma trận:

4	3
1	2

- $n = 3$ in ra ma trận:

7	6	5
8	9	4
1	2	3

- $n = 4$ in ra ma trận:

10	9	8	7
11	16	15	6
12	13	14	5

1	2	3	4
---	---	---	---

Bài tập 3: nhập vào từ bàn phím một xâu văn bản (các ký tự ASCII không là ký tự điều khiển, có độ dài không vượt quá 255) là tên một người. Hãy viết chương trình thực hiện đếm số lần xuất hiện của các ký tự trong xâu

Bài tập 4: Viết chương trình nhập vào một số N hãy in ra màn hình tất cả các số nguyên tố và số hoàn hảo nhỏ hơn N.

Chương 3, 4 và 5: Pointer, Reference and Function

Bài tập 5: Viết chương trình nhận một số thập phân từ bàn phím và chuyển nó thành dạng số ở cơ số bất kỳ.

Yêu cầu: Chương trình có khả năng nhận tham số bằng hai cách một là trên dòng lệnh hai là khi chương trình chạy.

Bài tập 6:

a) Viết chương trình nhập vào hai ma trận, tính tổng, tích của chúng và hiển thị kết quả lên màn hình.

b) Tìm phần tử nhỏ nhất ở mỗi hàng và phần tử lớn nhất trong các phần tử nhỏ nhất đó.

Yêu cầu:

- Việc nhập ma trận được thực hiện bằng các hàm với các mảng nguyên, thực
- Việc tìm phần tử nhỏ nhất và lớn nhất cũng được thực hiện bởi các hàm

Bài tập 7:

Trò chơi Puzzle là một trò chơi xếp các số trong một bảng hình vuông theo một thứ

tự nhất định chẳng hạn với một hình vuông 9 ô:

1	3	6
8	5	
2	4	7

Thì nhiệm vụ của người chơi là phải sắp xếp lại hình vuông trên sao cho nó có dạng:

1	2	3
4	5	6
7	8	

Hãy viết chương trình cài đặt trò chơi trên.

Yêu cầu:

- a. Có thể sử dụng giao diện dạng text hoặc đồ hoạ tùy ý, miễn là dễ thao tác cho người chơi
- b. Luật chơi là các chữ số bên cạnh ô trống có thể được chuyển sang ô trống

Bài tập 8:

Viết chương trình nhập vào một dãy các số (nguyên hoặc thực)

a) Hãy đưa ra số lớn nhất và nhỏ nhất trong dãy đó.

b) Hãy đưa ra phần tử lớn thứ k trong dãy số đó.

Dữ liệu được nhập vào từ file text có tên là dat25.txt trong thư mục input. File

dat25.txt có format như sau: mỗi dòng là một dãy các số số cuối cùng là một số nguyên

dương (k) để phục vụ cho phần b của bài toán. Chương trình sẽ đọc từng dòng file input và in kết quả ra màn hình cho tới hết.

Chương 6, 7, 8 : Object và Class

Bài tập 9: Lớp số nguyên lớn BigInt

Xây dựng lớp số nguyên lớn (sử dụng mảng hoặc chuỗi ký tự) với các phương thức sau:

- a. Cấu tử mặc định, số có giá trị bằng 0
- b. Cấu tử có tham số là một số nguyên kiểu int hoặc long.
- c. Cấu tử có tham số là một chuỗi.
- d. Cấu tử copy
- e. Hủy tử
- f. Các hàm toán tử gán, +, -, *, /, %, so sánh ==, >, <, <=, >=, !=.
- g. Hàm tính giai thừa
- h. Hàm kiểm tra xem một số nguyên lớn có là số nguyên tố hay không.
- i. Hàm toán tử xuất nhập.

Sử dụng lớp BigInt vừa xây dựng nhập vào 2 số nguyên và minh họa các phương thức đã được xây dựng của lớp.

Bài tập 10: Lớp đa thức Polynomial

Xây dựng lớp đa thức với các hệ số là các số thực với các phương thức sau:

- a. Cấu tử mặc định gán đa thức bằng 0.
- b. Cấu tử có tham số là một mảng hệ số và bậc của đa thức.

- c. Hủy tử
- d. Hàm toán tử nhập xuất
- e. Các hàm toán tử cộng, trừ, nhân hai đa thức
- f. Các hàm toán tử so sánh ==, !=
- g. Hàm tính giá trị của đa thức với giá trị biến là tham số của hàm.

Sử dụng lớp vừa cài đặt, nhập vào 2 đa thức và minh họa tất cả các phương thức đã xây dựng.

Bài tập 11: Lớp Stack

Xây dựng lớp stack (hoặc sử dụng thư viện STL) chứa các ký tự, sử dụng lớp stack

đó để kiểm tra xem:

Một chuỗi có là đối xứng hay không. Ví dụ chuỗi “abababa” thì kết quả là đúng, chuỗi “abd” thì kết quả là sai.

Kiểm tra các dấu đóng và mở ngoặc của một biểu thức có là hợp lệ hay không. Ví dụ nếu các dấu đóng, mở ngoặc là $((()))$ thì kết quả là đúng, còn nếu chuỗi là $(())()$ thì kết quả là sai.

Chương 9, 10, 11: Encapsulation, Inheritance và Polymorphism

Bài tập 12:

Hãy xây dựng cấu trúc dữ liệu bản mẫu danh sách liên kết (cài đặt bằng con trỏ) và kế thừa để xây dựng các cấu trúc dữ liệu ngăn xếp và hàng đợi ưu tiên.

Bài tập 13:

Có 3 lớp câu hỏi trắc nghiệm: loại 1 có các trường id, nội dung câu hỏi, các lựa chọn A, B, C, D và đáp án (là 1 trong 4 lựa chọn A, B, C, D), loại 2 có các trường id, nội dung câu hỏi, hình ảnh, các lựa chọn A, B, C, D và đáp án (là 1 trong 4 lựa chọn A, B, C, D), loại 3 có các trường id, nội dung câu hỏi, hình ảnh 1, hình ảnh 2, các lựa chọn A, B, C, D và đáp án (là 1 trong 4 lựa chọn A, B, C, D). Đề thi có tổng số m câu hỏi trong đó m1 câu được chọn ngẫu nhiên từ mảng n1 câu hỏi loại 1, m2 câu được chọn ngẫu nhiên từ mảng n2 câu hỏi loại 2 và m3 câu được chọn ngẫu nhiên từ mảng n3 câu hỏi loại 3 ($m1 + m2 + m3 == m$).

Hãy viết chương trình nhập vào một mảng gồm có: n1 câu hỏi loại 1, n2 câu hỏi loại 2, n3 câu hỏi loại 3, sau đó nhập các tham số m1, m2, m3, m và đưa ra các câu hỏi sẽ có trong đề thi.

Yêu cầu kỹ thuật: Chỉ sử dụng một mảng để lưu 3 loại câu hỏi, sử dụng mảng động và kỹ thuật đa thể (Polymorphism), các câu hỏi được chọn vào đề thi phải khác nhau

Chương 12: Nhập xuất File

Bài tập 14: Viết chương trình nhập một chuỗi từ bàn phím, viết các hàm thực hiện các công việc sau:

- Đưa ra độ dài chuỗi (hàm `length`).
- Chuẩn hóa sao cho giữa hai từ chỉ có một dấu cách, các dấu cách ở đầu và cuối được bỏ đi (hàm `trim`).
- Chuyển các ký tự thường thành các ký tự hoa (hàm `toUpperCase`).
- Chuyển các ký tự hoa thành các ký tự thường (hàm `toLowerCase`).
- Đảo ngược chuỗi (hàm `reverse`).
- Hàm `sub()` cho phép lấy một chuỗi con có độ dài là một tham số của hàm bắt đầu từ một vị trí nào đó của chuỗi.
- Hàm `strchr()` cho phép thay thế tất cả các ký tự của chuỗi bằng một ký tự khác.
- Hàm `pos()` cho biết vị trí xuất hiện đầu tiên của một chuỗi con trong một chuỗi khác.
- Sau khi gọi tới các hàm yêu cầu đưa ra kết quả tương ứng.

Bài tập 15: Viết chương trình đọc một mảng các số nguyên từ file `dat33.txt` trong thư mục `input` và viết các hàm sau:

- Hàm `reverse` đảo ngược mảng.
- Hàm `max` trả về giá trị lớn nhất của mảng.
- Hàm `sort` sắp xếp mảng theo thứ tự tăng dần
- Sau khi gọi tới các hàm yêu cầu đưa kết quả ra màn hình.

Chương 14: Template

Bài tập 16:

Hãy xây dựng cấu trúc dữ liệu bản mẫu hàng đợi (Queue) theo hai cách:

Cài đặt bằng con trỏ và cài đặt bằng mảng.

Bài tập 17:

Xây dựng lớp bản mẫu `DataSet` chứa một mảng `n` đối tượng thuộc lớp bản mẫu với các phương thức:

+ khởi tạo : nhập một mảng `n` phần tử kiểu bản mẫu

+ `show`: hiển thị đúng `pNo` phần tử của mảng.

+ `next`: hiển thị `pNo` phần tử tiếp theo bắt đầu từ vị trí hiện tại và tiến lên `pNo` vị trí.

+ `prev`: hiển thị `pNo` phần tử đứng trước vị trí hiện tại và lùi lại `pNo` vị trí

+ `begin`: trở về vị trí đầu tiên và hiển thị `pNo` phần tử từ vị trí đầu tiên.

+ `end`: hiển thị trang cuối cùng của mảng

+ `showAll`: hiển thị tất cả các phần tử của mảng

+ `sort`: sắp xếp mảng tăng dần

+ clear: xóa toàn bộ mảng

+ các hàm cấu tử và hủy tử nếu cần thiết.

Viết hàm main() minh họa các hàm của lớp bản mẫu đã cài đặt với kiểu dữ liệu số nguyên, chuỗi (String).