

TÀI LIỆU THỰC HÀNH KỸ THUẬT LẬP TRÌNH (C/C++)

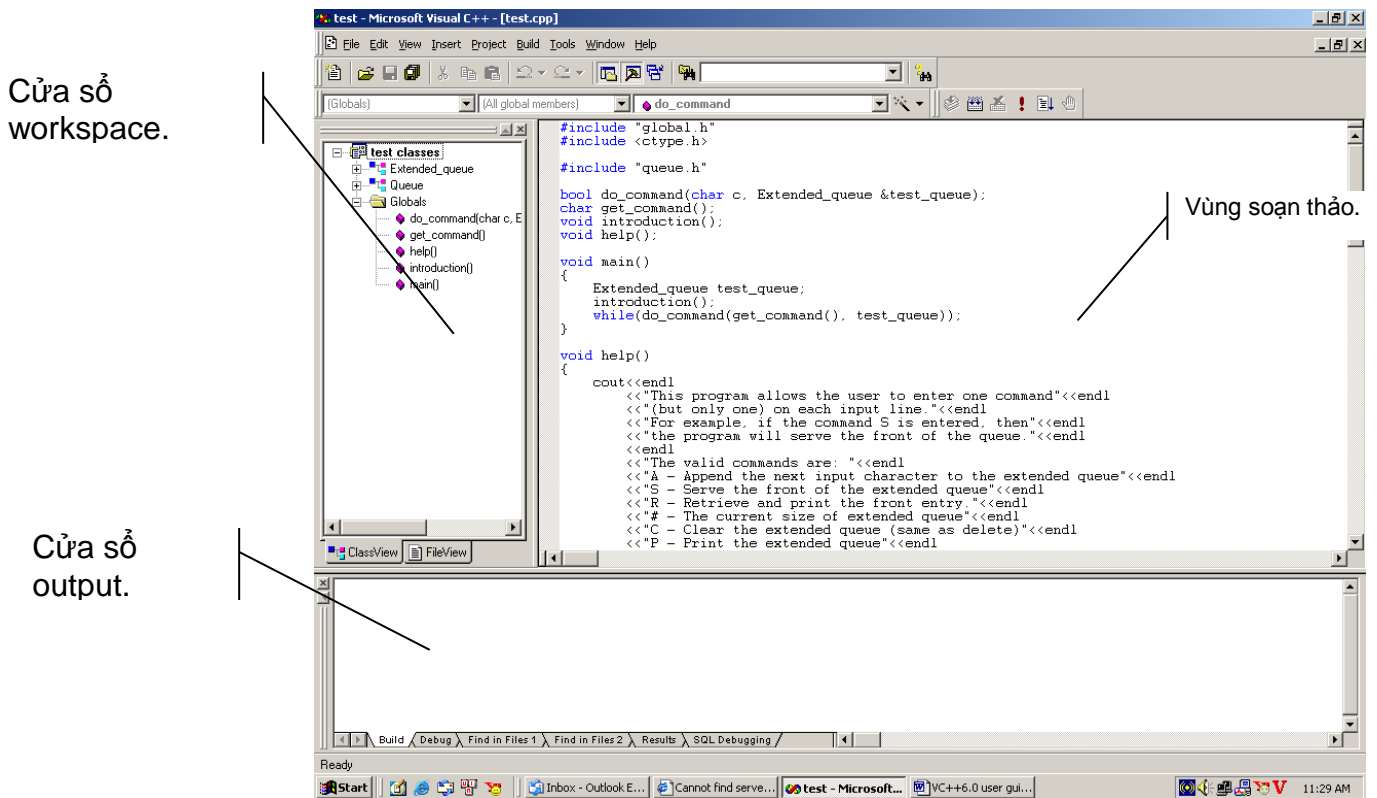
I. HƯỚNG DẪN CÁC BÀI THỰC HÀNH

Để các buổi thực hành thực sự có hiệu quả, đáp ứng nội dung học phần, sinh viên cần chú ý các hướng dẫn sau:

1. Nắm vững nội dung lý thuyết liên quan
2. Đọc kỹ các lưu ý
3. Chuẩn bị bài tập ở nhà:
 - + Yêu cầu bắt buộc sinh viên phải viết ra giấy các bài trước khi thực hành.
 - + Trong khi chuẩn bị bài, sinh viên tự mình làm; sau đó mới xem hướng dẫn và tham khảo các tài liệu.
4. Giảng viên sẽ chấm điểm của tất cả các buổi thực hành.
5. Sinh viên sử dụng Dev C++/Visual C++6.0/Visual Studio2010 để thực hành.

II. GIỚI THIỆU NNLT VISUAL C++6.0

Visual C++ 6.0 là phần mềm lập trình cho ngôn ngữ C++ với nhiều tính năng và hỗ trợ tốt. Một project trong VC++ chỉ dùng để lập trình cho một ứng dụng duy nhất. Để làm ví dụ thì ta có thể tạo ra nhiều project khác nhau cho những ví dụ khác nhau, hoặc đơn giản là sửa đổi đoạn chương trình trong hàm main.

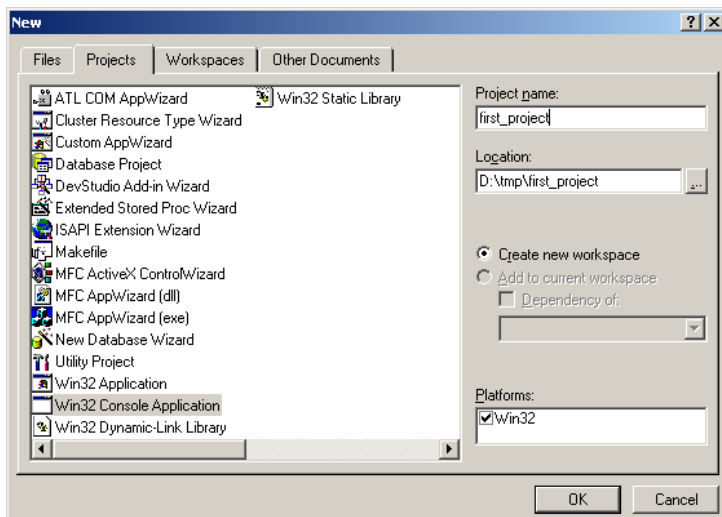


Giao diện Visual C++ 6.0

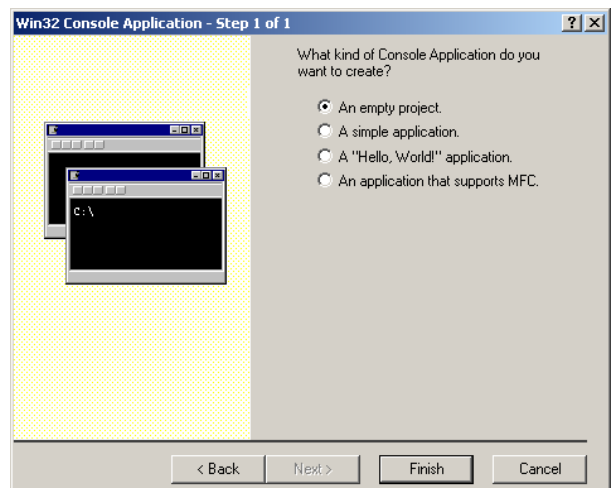
Các bước tạo ra một project đơn giản

- **Bước 1:** Tạo một project đơn giản
 - o Chọn File -> New, chọn tab Project (hình 2)
 - o Chọn *Win32 Console Application*
 - o Gõ tên của project muốn tạo vào mục *Project name*
 - o Chọn thư mục cần lưu trữ trong mục *Location*. Chú ý, sau khi chọn thư mục xong thì phần mềm tự động thêm tên của project vào để thành tên đầy đủ của thư mục chứa project này. Toàn bộ các tập tin của project này đều nằm trong thư mục đã chỉ ra.
 - o Nhấn nút OK
 - o Một hội thoại khác xuất hiện (Win32 Console Application – Step 1 of 1) để giúp ta tạo ra một project bước ban đầu. Để cho đơn giản ta chọn *An empty project* (hình 3).
 - o Nhấn nút *Finish* để tạo ra một project rỗng.
- **Bước 2:** Tạo một tập tin header *MyFirst.h*
 - o Chọn File -> New, chọn tab Files (hình 4)
 - o Chọn C/C++ Header Files
 - o Gõ tên tập tin với phần mở rộng là *.h* vào mục *File name*. Chú ý rằng dấu đánh dấu ở mục *Add to project* cho biết là tập tin mới tạo này sẽ được thêm vào trong project có tên được hiển thị lên đó. Nếu tên project mà khác thì phải chọn lại trong danh sách.

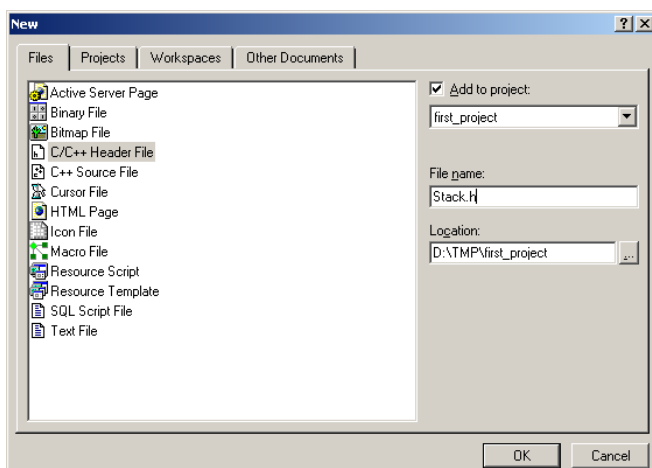
- Chú ý đến mục *Location* để đảm bảo là tập tin mới tạo ra cũng sẽ ở trong cùng thư mục với project vừa tạo ra.
 - Nhấn nút OK và bắt đầu soạn thảo
- **Bước 3:** Tạo một tập tin nguồn *MyFirst.cpp*
- Chọn File -> New, chọn tab Files (hình 5)
 - Chọn C/C++ Source Files
 - Gõ tên tập tin với phần mở rộng là **.cpp** vào mục *File name*.
 - Nhấn OK và bắt đầu soạn thảo.
- **Bước 4:** Tạo tập tin nguồn dùng thực thi *MyMain.cpp*
- Tạo một tập tin nguồn
 - Soạn thảo hàm chính của chương trình là **main**
- **Bước 5:** Thực thi chương trình
- Chọn Build -> Build *first_project.exe*
 - Chọn Build -> Execute *first_project.exe*
 - Chú ý là tên của project đang tạo ra sẽ được thay thế vào các chữ in nghiêng ở trên



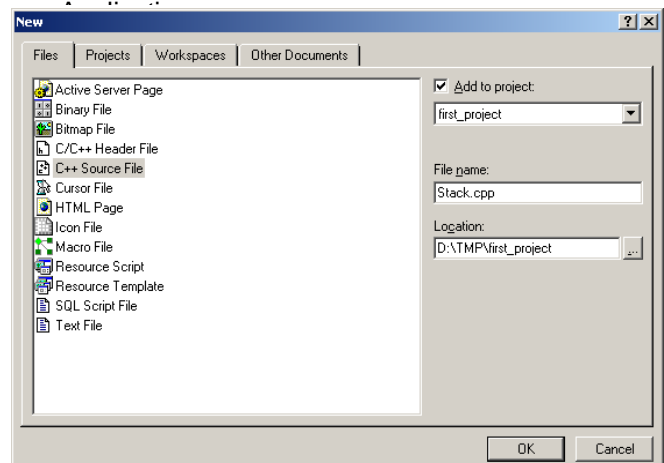
Hình 2. Tạo một project mới



Hình 3. Chọn kiểu project cho Win32 Console



Hình 4. Tạo một tập tin header



Hình 5. Tạo một tập tin nguồn

III. BÀI THỰC HÀNH SỐ 1

Làm quen môi trường lập trình

1. Mục tiêu

- Sử dụng môi trường soạn thảo và chạy một chương trình C++ đơn giản
- Hoàn thành cài đặt bài tập ở nhà và bài tập tại lớp

2. Yêu cầu (trước khi đến LAB)

- Đọc tài liệu hướng dẫn sử dụng Visual C++ 6.0
- Ôn lại kiến thức về NNLT C/C++: hằng, biến, kiểu dữ liệu, biểu thức, đối tượng xuất nhập: cout, cin...
- Hiểu rõ toán tử: +, -, *, /, %, ++, --, +=, -=, *=, /=, &&, ||, !, !=, >, <, <=, >=, ... và độ ưu tiên.
- Làm đầy đủ ra giấy các bài tập trong mục 4.

3. Lưu ý:

- Ngôn ngữ C/C++ phân biệt chữ hoa và chữ thường (khác với Pascal, VB...).
- Mỗi chương trình có duy nhất 1 hàm main. Khi chạy chương trình, hàm main là nơi đầu tiên được thực hiện.
- Mỗi khi có mở ngoặc thì phải có đóng ngoặc. vd: {...} và (...)
- Các dòng lệnh phải kết thúc bằng dấu chấm phẩy ";"
- Các dòng lệnh, các cặp {...} cùng cấp nên đóng thẳng cột để chương trình được rõ ràng.

4. Các bài tập:

Câu 1. Cài đặt bài tập Chương 1 (Phần lý thuyết).

Câu 2. Cho số tự nhiên n. Sử dụng các toán tử / và % viết chương trình xác định:

- a) chữ số đầu tiên
- b) chữ số hàng đơn vị
- c) tổng các chữ số

Câu 3. ☒ Viết chương trình đổi số giây n ra giờ (h), phút (m), giây (s). Ví dụ, với n = 15020 (giây) thì h=4, m=10, s=20.

Câu 4. ☒ Viết chương trình tính và in ra giá trị các biểu thức sau với 4 số lẻ:

a) $\sqrt{2 + \sqrt{2 + \sqrt{2}}}$

b) $\sqrt[5]{5 + \sqrt[4]{4 + \sqrt[3]{3 + \sqrt[2]{2}}}}$

c)
$$\frac{1}{3 + \frac{1}{3 + \frac{1}{3}}}$$

d) $S_2 = \frac{1}{1^2} + \frac{1}{3^2} + \dots + \frac{1}{(2n-1)^2}$

e) $S_3 = \frac{\sqrt{1+2+\dots+n}}{\sqrt[3]{1+\sqrt[3]{2}+\dots+\sqrt[3]{n}}}$

$$f) S_4 = \frac{1}{1!} + \frac{1}{2!} + \dots + \frac{1}{n!}$$

Câu 5. ☒ Nhập a, b, c là độ dài 3 cạnh của tam giác (thỏa mãn tổng 2 cạnh lớn hơn cạnh còn lại). Tính chu vi, diện tích, độ dài 3 đường cao, 3 đường trung tuyến, 3 đường phân giác, bán kính đường tròn nội tiếp, ngoại tiếp lần lượt theo các công thức sau:

$$C = 2p = a + b + c; \quad S = \sqrt{p(p-a)(p-b)(p-c)};$$

$$h_a = \frac{2S}{a}; \quad m_a = \frac{1}{2} \sqrt{2b^2 + 2c^2 - a^2}; \quad g_a = \frac{2}{b+c} \sqrt{2bcp(p-a)};$$

$$r = \frac{S}{p}; \quad R = \frac{abc}{4S}$$

Câu 6. Tính giá trị của hàm $f(x) = \sqrt[3]{\sin x + 3 \cos x}$ tại $x = \frac{\pi}{2}, \frac{\pi}{3}$.

Câu 7. Lực hấp dẫn giữa hai vật thể có khối lượng M_1 và M_2 cách nhau một khoảng r được tính theo công thức $F = \gamma \frac{M_1 M_2}{r^2}$

trong đó $\gamma = 6.673 \times 10^{-8} \text{ cm}^2/\text{gs}^2$ là hằng số hấp dẫn.

Viết chương trình tính lực hấp dẫn của hai vật thể có các khối lượng và khoảng cách giữa chúng được nhập từ bàn phím.

Câu 8. ☒ Viết chương trình hiển thị lên màn hình các câu hỏi Yes/No như sau trong bài toán phỏng vấn LTV:

- [1 điểm] Bạn chưa có gia đình (Y/N)?
- [2 điểm] Tuổi dưới 30 (Y/N)?
- [3 điểm] Đã tốt nghiệp ĐH (Y/N)?
- [2 điểm] Có chứng chỉ tiếng Anh quốc tế (Y/N)?
- [2 điểm] Có chứng chỉ LTV (Y/N)?

Trong mỗi câu hỏi, nếu trả lời 'Y' được điểm tương ứng, là 0 điểm nếu trả lời 'N'. Xét tuyển như sau, Nếu tổng điểm:

- Từ 8 đến 10: Xin chúc mừng!
- Từ 4 đến 7: Về học thêm, hẹn gặp lại!
- Còn lại: Mong thông cảm!

Câu 9. ☒ Viết chương trình thực hiện: Kiểm tra tính hợp lệ ngày, tháng, năm nhập từ bàn phím.

- Cho biết tháng nhập có bao nhiêu ngày.
- Cho biết ngày hôm sau của ngày đã nhập là ngày nào.
- Cho biết ngày hôm trước của ngày đã nhập là ngày nào. VD: Nếu ngày nhập là 1/12/2009 thì ngày trước đó là 30/11/2009.

Gợi ý:

- Tháng 1, 3, 5, 7, 8, 10, 12: 31 ngày
- Tháng: 3, 4, 6, 9, 11: 30 ngày
- Tháng 2: Nhuận/Không nhuận: 29/28 ngày

- Năm nhuận là (những năm chia hết cho 400) hoặc (chia hết cho 4 và không chia hết cho 100 hoặc 400). Ví dụ 2000 là năm nhuận nhưng 2100 không phải là năm nhuận.

Câu 10. Viết chương trình kiểm tra phương trình

$$2 \sin x + 3 \cos x + 1 = 0$$

có nghiệm trong các khoảng nào sau đây: $(0, \frac{\pi}{2})$, $(\frac{\pi}{2}, \pi)$, $(\pi, \frac{3\pi}{2})$?

Câu 11. Tìm và in lên màn hình tất cả các số nguyên trong phạm vi từ 10 đến 99 sao cho tích của 2 chữ số bằng 2 lần tổng của 2 chữ số đó.

Câu 12. Nhập vào một số nguyên n, in ra tất cả các ước số của n.

Câu 13. ☒ Tìm các số hoàn hảo nhỏ hơn 1000. Một số hoàn hảo là số có tổng các ước số thực sự của nó bằng chính nó. Ví dụ: 6 là số hoàn hảo vì tổng các ước $1+2+3 = 6$.

Câu 14. Viết chương trình tìm và đếm số nghiệm nguyên không âm của phương trình

$$x + y + z = 30$$

Câu 15. Nhập vào số tự nhiên N, tìm tất cả các số tự nhiên q sao cho N chia hết cho q^2 và không chia hết cho q^3 .

Câu 16. Tìm các số Armstrong gồm 3 chữ số. Biết rằng, số Armstrong gồm n ký số thì tổng các lũy thừa bậc n của các ký số bằng chính nó.

$$\text{Ví dụ: } 153 = 1^3 + 5^3 + 3^3 = 1 + 125 + 27 = 153.$$

Câu 17. Tìm các số tự nhiên có 3 chữ số sao cho chữ số hàng trăm bằng chữ số hàng đơn vị.

Câu 18. ☒ Nhập vào một số nguyên dương gồm nhiều chữ số. Tính số chữ số, tổng các chữ số, và tính trung bình cộng các chữ số của số đã nhập.

IV. BÀI THỰC HÀNH SỐ 2

Kỹ thuật tổ chức chương trình

1. Mục tiêu

- Nắm được phương pháp phân tích Top-down
- Hiểu và ứng dụng các nguyên tắc lập trình
- Hiểu và ứng dụng về các phương pháp tối ưu mã nguồn

2. Yêu cầu (trước khi đến LAB)

- Đọc kỹ các bài giảng để nắm được nội dung lý thuyết, đáp ứng được mục tiêu.
- Chú ý đến kỹ thuật nhỏ nhất trong lập trình.
- Làm đầy đủ ra giấy các bài tập trong mục 3.

3. Các bài tập

Câu 1. Cài đặt Bài tập 4 phần lý thuyết.

Câu 2. Cho biết kết quả khi chạy đoạn chương trình sau:

```
int main()
{
    int k, N=8;
    for (k = 2; k <= N; k += 2)
        cout << k;

    k++;
    cout << k;
    return 0;
}
```

Câu 3. Cho biết kết quả khi chạy đoạn chương trình sau:

```
#include <iostream>
int main ()
{
    int x = 3, counter = 0;
    while ((x-1))
    {
        ++counter;
        x--;
    }
    cout << counter;
    return 0;
}
```

Câu 4. ☒ Có các loại giấy bạc 1đ, 5đ, 10đ và 20đ. Nhập số tiền cần rút m, hãy lập chương trình để tìm tất cả các phương án rút tiền có thể. Tìm phương án tốt nhất (có số tờ giấy bạc ít nhất).

Câu 5. Viết chương trình in ra bảng cửu chương từ 1 đến 10

Câu 6. ☒ Nhập số nguyên n ($0 < n < 20$), lập trình in ra các hình gồm ký tự '*' như sau (ví dụ n=9):

```
*
* *
* * *
* * * *
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
```

```
*
* *
* * *
* * * *
* * * * *
* * * *
* * *
* *
*
```

```
*           *
*           *
*           *
*           *
*           *
*           *
*           *
*           *
*           *
*           *
```

```
* * * * *
* *           *
* *           *
* *           *
* *           *
* *           *
* *           *
* *           *
* *           *
* *           *
```

Câu 7. Hiển thị số nguyên trong khoảng [A, B] chia hết cho là bội số chung của 2 số nguyên x, y với A, B, x, y được nhập vào từ bàn phím

Câu 8. Bài toán cổ: vừa gà vừa chó bó lại cho tròn, đếm đủ 100 chân. Hỏi có mấy gà và mấy con chó, biết tổng số con là 36.

Câu 9. ☒ Viết chương trình tính giá trị của các giá trị sau:

a) $n! = 1 * 2 * \dots * n$

b) $\frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \dots + \frac{(-1)^n}{2n+1}$

c) $e = 1 + \frac{1}{1!} + \frac{1}{2!} + \dots + \frac{1}{n!}$

Câu 10. ☒ Viết chương trình nhập số tự nhiên n và in giá trị của:

a) $\sum_{i=1}^n (2i-1)$

b) $\sum_{i=1}^n \frac{1}{i}$

c) $\sum_{i=1}^n \frac{1}{i(i+1)}$

d) $\sum_{i=1}^n i^3$

e) $\sum_{i=1}^n i(i+1)$

f) $\sum_{i=1}^n (-1)^{i+1} i$

g) $\sqrt{2 + \sqrt{2 + \dots + \sqrt{2}}} \quad (n \text{ dấu căn})$

h) $\sqrt{3 + \sqrt{6 + \dots + \sqrt{3(n-1) + \sqrt{3n}}}}$

Câu 11. Viết chương trình nhập số tự nhiên n, và in giá trị sau:

- a) $\frac{1}{1} \cdot \frac{3}{2} \cdot \frac{5}{3} \cdots \frac{2n-1}{n}$
- b) $(1 + \frac{1}{2})(1 + \frac{1}{3}) \cdots (1 + \frac{1}{n})$
- c) $\frac{1^2 + 2^2 + \dots + n^2}{\sqrt{1 + 2 + \dots + n}}$
- d) $\frac{\sqrt{1} + \sqrt{2} + \dots + \sqrt{n}}{1^2 + 2^2 + \dots + n^2}$

Câu 12. Nhập vào số N . Kiểm tra N có phải là số chính phương không?

Gợi ý: Nếu tồn tại số a để $N = a^2$ thì N là số chính phương. VD: 9 là số chính phương

Câu 13. Tính lãi suất tiết kiệm.

Khi gửi vào ngân hàng một số tiền P , với lãi suất thường niên được biểu diễn qua số thực R (ví dụ, lãi suất 7.5% sẽ được biểu diễn là 0.075), số tiền thu được sau khi gửi N năm sẽ là $P \cdot (1 + R)^N$. Viết chương trình đọc các giá trị P , R , N từ bàn phím và in số tiền thu được tương ứng ra màn hình theo từng năm.

Câu 14. ☒ Viết chương trình thực hiện:

- Đợi người sử dụng nhập vào một số tự nhiên từ bàn phím;
- Đưa ra kết luận số đó có phải số nguyên tố hay không;
- Chương trình hỏi người sử dụng: “Bạn muốn nhập số khác (C/K)?”. Nếu người dùng nhập phím ‘k’ hoặc ‘K’ thì chương trình kết thúc, ngược lại quá trình trên sẽ được lặp lại.

Câu 15. ☒ Viết chương trình in ra n số nguyên tố đầu tiên ($0 < n < 1000$).

Câu 16. Viết chương trình nhập một số nguyên dương n , tìm chữ số lớn nhất của nó. Ví dụ, với $n=2013$ chữ số lớn nhất là 3.

Câu 17. Viết chương trình nhập một số tự nhiên n . Cho biết

- tổng các chữ số của n
- In bằng chữ của n .

Ví dụ. *Input:* 5247

Output: 18

Nam hai bon bay

Câu 18. ☒ Viết chương trình tìm n nguyên dương lớn nhất thỏa mãn bất phương trình:

- $n^2 + 2n < 100$
- $500 - 2^n > 0$

V. BÀI THỰC HÀNH SỐ 3

Hàm, mảng một chiều

1. Mục tiêu

- Nắm được phương pháp lập trình cấu trúc
- Hiểu các cách truyền đối số; biến toàn cục và cục bộ
- Hiểu và ứng dụng cách thức module hóa và xây dựng hàm

2. Yêu cầu (trước khi đến LAB)

- Đọc kỹ các bài giảng để nắm được nội dung lý thuyết, đáp ứng được mục tiêu.
- Làm đầy đủ ra giấy các bài tập trong mục 3.

3. Các bài tập

Sử dụng hàm để giải quyết các bài toán sau

Câu 1. Nhập số tự nhiên n và số thực x . In ra màn hình giá trị các tổng, tích sau

- a) $\frac{1}{x} + \frac{1}{x(x+1)} + \dots + \frac{1}{x(x+1)\dots(x+n)}$
- b) $\sin x + \sin^2 x + \dots + \sin^n x$
- c) $x(x+1)\dots(x+n)$

Câu 2. ☒ Nhập x, ε . Tính và in giá trị của

$$e^x = 1 + \frac{x}{2!} + \frac{x^2}{3!} + \dots + \frac{x^n}{n!} \quad \text{cho đến khi} \quad \left| \frac{x^{n+1}}{(n+1)!} \right| < \varepsilon$$

Câu 3. Viết chương trình nhập vào số thực a thỏa $0 < a < 5$. Xác định số tự nhiên n bé nhất thỏa mãn:

$$1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}.$$

Câu 4. Viết chương trình xác định số tự nhiên n để hiệu hai phần tử liên tiếp của dãy

$$1 + \frac{1}{2!} + \frac{1}{3!} + \dots + \frac{1}{n!}.$$

bé hơn số ε , ($0 < \varepsilon < 1$) nhập từ bàn phím.

Câu 5. Tính giá trị của biểu thức tương ứng dưới đây, với $n > 0$ được nhập vào từ bàn phím.

$$S1 = 1 - \frac{1}{1^2 + 2^2} + \frac{1}{1^2 + 2^2 + 3^2} - \dots + \frac{(-1)^{n+1}}{1^2 + 2^2 + 3^2 \dots + n^2}$$

$$S2 = 1 - \frac{2}{1!} + \frac{2^2}{2!} - \frac{2^3}{3!} + \dots + \frac{(-2)^n}{n!}$$

$$S3 = \underbrace{\sqrt{x + \sqrt{x + \sqrt{x + \dots + \sqrt{x + \sqrt{x}}}}}}_n$$

$$S4 = \frac{1}{2} - \frac{1}{4} + \frac{1}{6} - \frac{1}{8} + \dots + (-1)^{n+1} \frac{1}{2n}$$

$$S5 = 1 + \frac{1}{1^2 + 2^2} + \frac{1}{1^2 + 2^2 + 3^2} + \dots + \frac{1}{1^2 + 2^2 + 3^2 \dots + n^2}$$

$$S6 = 1 - \frac{1}{1+2} + \frac{1}{1+2+3} - \dots + \frac{(-1)^{n+1}}{1+2+3+\dots+n}$$

$$S7 = 1 + \frac{2}{1!} + \frac{2^2}{2!} + \frac{2^3}{3!} + \dots + \frac{2^n}{n!}$$

$$S8 = \frac{1}{1!} - \frac{1}{2!} + \frac{1}{3!} - \frac{1}{4!} + \dots + (-1)^{n+1} \frac{1}{n!}$$

Câu 6. ☒ Viết chương trình nhập vào 2 số nguyên và kiểm tra chúng có nguyên tố cùng nhau? Hai số nguyên m và n được gọi là nguyên tố cùng nhau nếu $\text{USCLN}(m, n) = 1$.

Câu 7. ☒ Viết chương trình nhập vào một mảng số nguyên, hãy xuất ra màn hình, mỗi kết quả trên một dòng:

- Gồm các số lẻ, tổng cộng có bao nhiêu số lẻ.
- Gồm các số chẵn, tổng cộng có bao nhiêu số chẵn.
- Phần tử lớn nhất, nhỏ nhất của mảng.
- Tổng của các phần tử của mảng.
- Trung bình cộng các phần tử
- Phần tử âm lớn nhất, phần tử dương nhỏ nhất của mảng.
- Tính đối xứng của mảng
- Tổng của các phần tử là số nguyên tố trong mảng.
- Số phần tử là số nguyên tố trong mảng.
- Số nguyên tố đầu tiên, sau cùng trong mảng

Câu 8. Viết chương trình nhập n phần tử số nguyên. Nhập phần tử cần tìm kiếm X. Nếu trong n phần tử đã nhập có X thì báo "tìm thấy", "số lần tìm thấy" và "các vị trí tìm thấy", ngược lại báo "không tìm thấy".

Câu 9. ☒ Viết chương trình nhập một dãy số thực đôi một khác nhau. Xác định vị trí của phần tử lớn nhất và nhỏ nhất trong dãy, sau đó hoán đổi vị trí 2 phần tử này cho nhau, các vị trí khác không thay đổi.

Ví dụ: Input: 3, 4, 1, 5, 9, 12, 10, 2

Output: 3, 4, 12, 5, 9, 1, 10, 2

Câu 10. Viết chương trình nhập một dãy số bất kỳ và một số M. Kiểm tra xem có 2 phần tử liên tiếp nào có tổng bằng M không?

Câu 11. Viết chương trình nhập một dãy n số thực a_1, a_2, \dots, a_n . Kiểm tra dãy này có đơn điệu hay không (tăng hay giảm). Nếu là đơn điệu thì nhập vào số thực x và chèn x vào dãy đã cho sao cho dãy vẫn là đơn điệu.

Câu 12. ☒ Viết chương trình kiểm tra một dãy số nhập từ bàn phím được sắp xếp tăng (giảm) dần chưa? Nếu chưa, hãy sắp xếp tăng (giảm) dần dãy số đó.

Câu 13. Viết chương trình nhập hai dãy số thực a_1, a_2, \dots, a_n và b_1, b_2, \dots, b_n . Xác định

$$\text{a) } x = \sum_{i=1}^n a_i b_i \qquad \text{b) } y = \sum_{i=1}^n a_i b_{n-i+1}$$

Câu 14. Cho số tự nhiên n và các số thực $x, a_0, a_1, a_2, \dots, a_n$. Tính giá trị của đa thức:

$$P_n(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$$

Câu 15. Viết chương trình nhập hai dãy số thực tăng dần a_1, a_2, \dots, a_n và b_1, b_2, \dots, b_n . Hãy trộn 2 dãy trên thành một dãy tăng dần.

Câu 16. Viết chương trình nhập một dãy tọa độ thực (x_i, y_i) $i = \overline{1, n}$ và bán kính r của vòng tròn có tâm tại $(0,0)$. In các điểm rơi vào trong đường tròn (điểm tại biên của một đường tròn xem như không thuộc đường tròn đó).

Câu 17. Viết chương trình nhập n số nguyên lẻ a_1, a_2, \dots, a_n . Xây dựng dãy b_1, b_2, \dots, b_n được xác định như sau:

$$\begin{cases} b_1 = a_1, b_n = a_n \\ b_i = \frac{a_i + a_{i+1}}{2}, i = \overline{2, n-1} \end{cases}$$

Câu 18. Một dãy n số thực a_1, a_2, \dots, a_n gọi là dãy “zigzag” nếu $a_1 \leq a_2 \geq a_3 \leq a_4 \geq \dots$. Viết chương trình kiểm tra một dãy nhập vào có là dãy “zigzag” hay không?

Câu 19. Cho một dãy được xác định như sau:

$$\begin{cases} u_1 = 1 \\ u_n = u_{n-1} + 2, \quad n \geq 2 \end{cases}$$

Viết chương trình

- In n phần tử đầu tiên của dãy.
- Kiểm tra số M bất kỳ nhập từ bàn phím có thuộc dãy không?

Câu 20. *Tự sát tập thể (Josephus).* Một tàn quân gồm N người đã quyết định tự sát tập thể trong một hang núi bằng cách tự đứng trong một vòng tròn và giết người thứ M , thu hẹp hàng ngũ lại khi từng người lần lượt ngã khỏi vòng tròn. Xác định xem người nào là người chết cuối cùng (người này có thể thay đổi quyết định vào phút chót!).

Ví dụ: Với $N=9$ và $M=5$, thứ tự bị giết là 5, 1, 7, 4, 3, 6, 9, 2, 8.

Câu 21. Viết chương trình tạo một dãy chứa 10 số ngẫu nhiên là các số nguyên dương trong khoảng từ 1 đến 100 rồi in ra màn hình dãy số đó.

Câu 22. Viết chương trình giả lập đồng xu và tung n lần. Tính tần suất xuất hiện của các mặt.

Câu 23. ☒ Viết chương trình giả lập hạt xúc xắc và tung n lần. Tính tần suất xuất hiện của các mặt.

Câu 24. Viết chương trình tạo hai dãy a_1, a_2, \dots, a_{10} và b_1, b_2, \dots, b_{10} chứa các số ngẫu nhiên phân biệt từ 1 đến 100, trong đó dãy a_i gồm các số lẻ, dãy b_i gồm các số chẵn. In ra màn hình hai dãy trên.

Câu 25. Viết chương trình tính gần đúng nghiệm trong khoảng $[1, 2]$ sau n bước lặp của phương trình $x^3 - x - 1 = 0$ theo các phương pháp:

- Chia đôi
- Lặp đơn
- Dây cung
- Tiếp tuyến

VI. BÀI THỰC HÀNH SỐ 4

Mảng hai chiều và chuỗi ký tự

1. Mục tiêu

Sử dụng thành thạo mảng 2 chiều và chuỗi ký tự trong C++, bao gồm:

- Khai báo, truy cập mảng 2 chiều, chuỗi ký tự
- Giải thuật cơ bản trên ma trận.

2. Yêu cầu (trước khi đến LAB)

- Nắm vững kiến thức toán về ma trận
- Nắm vững các hàm xử lý xâu ký tự
- Làm đầy đủ ra giấy các bài tập trong mục 3.

3. Các bài tập

Câu 1. ☒ Viết chương trình nhập một ma trận vuông cấp n gồm các phần tử là số nguyên. In ra màn hình:

- ma trận vừa nhập
- giá trị và vị trí phần tử lớn nhất, nhỏ nhất
- tổng các phần tử trên đường chéo chính của ma trận
- tổng các phần tử trong tam giác trên của ma trận
- tổng các phần tử trong tam giác dưới của ma trận
- tổng các phần tử trong ma trận
- dòng có tổng lớn nhất
- cột có tổng nhỏ nhất
- kết quả kiểm tra ma trận có đối xứng qua đường chéo chính? đường chéo phụ?

Câu 2. ☒ Nhân 2 ma trận. Cho 2 ma trận A ($m \times n$) và B ($n \times p$). Tính ma trận $C = A \times B$, trong đó C có kích thước là $m \times p$.

Câu 3. Viết chương trình nhập ma trận kích thước $m \times n$. In ma trận chuyển vị của nó.

Câu 4. Một phần tử gọi là *điểm yên ngựa* của ma trận nếu nó là phần tử bé nhất của hàng chứa nó đồng thời là phần tử lớn nhất của cột chứa nó. Viết chương trình nhập một ma trận gồm các số nguyên và tìm các điểm yên ngựa của nó.

Câu 5. ☒ Viết chương trình tạo một ma trận vuông cấp n gồm các phần tử là số nguyên ngẫu nhiên 2 chữ số. Tính và in ra màn hình tổng của các phần tử lớn nhất trên mỗi dòng của ma trận.

Câu 6. Viết chương trình nhập một ma trận vuông cấp n gồm các phần tử 0 và 1. Xác định ma trận thu được là *ma trận thưa* hay không? Ma trận là thưa nếu có số phần tử 0 nhiều hơn số phần tử 1.

Câu 7. Viết chương trình nhập một ma trận vuông cấp n gồm các phần tử là số nguyên. In ma trận đã được sắp xếp sao cho trên mỗi dòng, các phần tử lớn nhất được hoán vị nằm trên đường chéo chính.

Ví dụ:

$$\text{Input: } \begin{bmatrix} 1 & 3 & 4 \\ 2 & 5 & 7 \\ 5 & 1 & 12 \end{bmatrix} \quad \text{Output: } \begin{bmatrix} 4 & 3 & 1 \\ 2 & 7 & 5 \\ 5 & 1 & 12 \end{bmatrix}$$

Câu 8. Viết chương trình nhập một ma trận vuông cấp n gồm các phần tử là số nguyên. In ma trận

- a) sắp xếp tăng dần trên tất cả các dòng.
- b) sắp xếp tăng dần theo đường xoắn ốc.

Câu 9. Viết chương trình xoay một ma trận $n \times n$ một góc 90/180 độ theo chiều kim đồng hồ, theo hai cách:

- a) Dùng ma trận phụ.
- b) Không dùng ma trận phụ.

Câu 10. ☒ Nhập một ma trận A vuông cấp n gồm các phần tử là số nguyên. Xác định ma trận B từ ma trận A bằng cách

- a) loại bỏ dòng i cột j .
- b) hoán vị hai cột i và j .

Câu 11. Bàn cờ vua quốc tế được xem như mảng hai chiều gồm các chỉ số trong khoảng từ 1 đến 8. Cho các số tự nhiên $1 \leq k, l, m, n \leq 8$. Viết chương trình kiểm tra

- a) Hai ô (k, l) và (m, n) có cùng màu với nhau không?
- b) Đặt hậu ở ô (k, l) có khống chế được ô (m, n) hay không?
- c) Như câu b) nhưng thay hậu bởi mã.

Câu 12. Bảng kết quả của một giải vô địch bóng đá được cho bởi ma trận vuông cấp n , trong đó các phần tử trên đường chéo chính bằng 0 còn các phần tử khác bằng 3, 1 hoặc 0 (tương ứng với số điểm của trận đấu: 3 thắng, 1 hoà, 0 thua).

- a) Hãy tìm số các đội có số trận thắng lớn hơn số trận thua.
- b) Hãy xác định chỉ số của đội không thua trận nào.
- c) Có hay không một đội có số trận thắng nhiều hơn một nửa số trận đấu.
- d) Tính số điểm của mỗi đội và sắp xếp thứ tự các đội giảm dần theo điểm.

Câu 13. Xây dựng các từ mới từ từ COMPUTER bằng cách hoán vị các ký tự của từ.

Câu 14. Nhập một chuỗi ký tự. In ra

- a) Từ đầu tiên và từ cuối cùng.
- b) Chiều dài của từ dài nhất.
- c) Số từ có chiều dài bé hơn 4.

Ví dụ: *Input: Truong Dai Hoc Nha Trang*

Output: Truong

Trang

6

3

Câu 15. Viết chương trình nhập một chuỗi ký tự và hai số tự nhiên n, m . Hoán đổi vị trí của hai từ thứ n và m cho nhau. Đưa ra màn hình chuỗi kết quả. Trường hợp, nếu n, m không phù hợp thì thông báo không đổi chỗ được.

Câu 16. ☒ Nhập chuỗi bất kỳ, sắp xếp các từ của chuỗi theo thứ tự alphabet.

Ví dụ: *Input:* *Xuan Ha Thu Dong*

Output: *Dong Ha Thu Xuan*

Câu 17. Viết chương trình nhập một chuỗi ký tự rồi in ra:

- Dòng 1: chuỗi ở dạng viết hoa.
- Dòng 2: chuỗi ở dạng viết thường.
- Dòng 3: chuỗi ở dạng không có khoảng trắng ở đầu, ở cuối; giữa 2 từ chỉ có 1 khoảng trắng.
- Dòng 4: chuỗi ở dạng ký tự đầu mỗi từ viết hoa, còn lại viết thường.

Câu 18. ☒ Nhập mật khẩu (không quá 10 ký tự). In ra "đúng" nếu là "nhatrang", "sai" nếu ngược lại. Chương trình cho phép nhập tối đa 3 lần. Nhập riêng rẽ từng ký tự (bằng hàm getch()) cho mật khẩu. Hàm getch() không hiện ký tự NSD gõ vào, thay vào đó chương trình chỉ hiện ký tự 'X' để che giấu mật khẩu. Sau khi NSD đã gõ xong (9 ký tự) hoặc đã Enter, chương trình so sánh xâu vừa nhập với "nhatrang", nếu đúng chương trình tiếp tục, nếu sai tăng số lần nhập (cho phép không quá 3 lần).

Câu 19. Với chuỗi ký tự S cho trước. Viết chương trình cho biết trong chuỗi S có bao nhiêu ký tự khác nhau và chúng xuất hiện bao nhiêu lần trong chuỗi S .

Câu 20. Không sử dụng các hàm có sẵn. Viết chương trình xóa N ký tự tại vị trí i trong chuỗi S .

Ví dụ. *Input:* *S = "Nha Trang"; i = 8; N = 2*

Output: *S = "Nha Tra"*

Câu 21. Nhà quân sự thiên tài của La Mã cổ đại Julius Cesar đã mã hoá văn bản bằng cách sau: Xếp bằng chữ cái thành vòng tròn, sau đó thay mỗi ký tự bằng ký tự sau nó n vị trí trong vòng tròn. Ví dụ với $n = 3$ thì 'news' được mã hoá thành 'qhzv'. Viết chương trình nhập số tự nhiên n , và một chuỗi ký tự (giả sử chỉ xét văn bản là ký tự tiếng Việt viết thường không dấu). Viết chương trình

- Mã hoá một chuỗi ký tự.
- Giải mã một chuỗi ký tự (xem như đã mã hoá).

Câu 22. Để nâng cao độ tin cậy khi truyền các bit thông tin, mỗi bit được lặp lại 3 lần. Ví dụ "001" được truyền thành "000111111". Khi nhận được thông tin, cứ một đoạn ba bit được giải mã thành 1 bit là số gấp trong đoạn đó ít nhất là hai lần. Chẳng hạn, nếu thông tin nhận được là "000110010011" thì thông tin giải mã là "0101". Cho chuỗi nhị phân là thông tin nhận được. Viết chương trình giải mã thông tin đó.

Câu 23. Nhập từ bàn phím hai chuỗi ký tự S và M . Có thể kiểm tra xem có thể nhận được M từ S bằng cách xoá đi một số ký tự của S hay không? Nếu được hãy hiển thị số thứ tự của các ký tự được giữ lại trong chuỗi S .

Câu 24. Viết chương trình cộng, trừ hai số nguyên có nhiều chữ số (dùng chuỗi).

Câu 25. Tín hiệu truyền đi là một dãy các số 0 và 1. Viết chương trình tạo ngẫu nhiên một chuỗi S là chuỗi nhị phân ngẫu nhiên dài n . In ra màn hình chuỗi S và số lượng các số 0,1.

Câu 26. ☒ Nhập xâu ký tự từ bàn phím, đếm xem trong xâu ký tự nào xuất hiện nhiều nhất.

Câu 27. Viết hàm tính độ dài xâu (không sử dụng thư viện). Viết chương trình nhập xâu từ bàn phím, sử dụng hàm vừa xây dựng đưa ra độ dài xâu.

Câu 28. Cho một xâu bất kỳ, không sử dụng các hàm thư viện về xâu, hãy xây dựng một hàm đổi tất cả các **chữ thường** thành **chữ hoa** (các ký tự khác giữ nguyên) và in cả hai ra màn hình. Viết một chương trình nhập một xâu bất kỳ từ bàn phím, sau đó sử dụng hàm đã xây dựng ở trên để in kết quả ra màn hình.

VII. BÀI THỰC HÀNH SỐ 5

Kiểu bản ghi và kiểu con trỏ

1. Mục tiêu

- Hiểu ý nghĩa và ứng dụng thành thạo con trỏ cấp 1 & 2
- Sử dụng tốt dữ liệu cấu trúc bản ghi

2. Yêu cầu (trước khi đến LAB)

- Đọc kỹ các bài giảng để nắm được nội dung lý thuyết, đáp ứng được mục tiêu.
- Làm đầy đủ ra giấy các bài tập trong mục 3.

3. Các bài tập

Câu 1. ☒ Cài đặt các ví dụ về kiểu cấu trúc và con trỏ phần lý thuyết.

Câu 2. ☒ Nhập mảng n sinh viên gồm các thông tin: tên, giới tính, điểm toán, điểm lý, điểm hoá. In danh sách (số thứ tự, tên, tổng điểm) các sinh viên nữ có thi lại theo thứ tự tăng dần của tổng điểm.

Câu 3. Cài đặt tương tự câu 2 nhưng sử dụng danh sách liên kết đơn.

Câu 4. ☒ Cho cấu trúc

```
struct thisinh{
    int        sbd;           //Số báo danh
    char        hoten[25];    //Họ và tên
    float       m1,m2,m3l;    //Điểm ba môn thi
    float       tong;         //Tổng điểm ba môn
} danhsach[100];
```

Viết chương trình (có sử dụng các hàm) để sắp xếp các thí sinh theo thứ tự giảm dần của tổng điểm, và in ra màn hình danh sách đã sắp.

Câu 5. ☒ Cho khai báo sau:

```
struct SinhVien
{
    char ho_ten[20];
    float diem;
}SV;
```

Viết chương trình thực hiện các chức năng sau:

- Nhập thông tin cho 10 sinh viên
- In danh sách ra màn hình
- In ra danh sách các sinh viên có điểm dưới 5.0
- Cho biết số sinh viên xếp loại xuất sắc.
- Sắp xếp thứ tự theo điểm trung bình tăng dần

Câu 6. Viết chương trình nhập thông tin tiêu thụ điện của khách hàng gồm: Tên khách hàng (kiểu chuỗi), chỉ số cũ (số nguyên), chỉ số mới (số nguyên), đơn giá (số nguyên), và xuất thông tin lên màn hình gồm tên khách hàng, tháng, số kWh tiêu thụ và số tiền phải trả.

Câu 7. Thông tin của nhân viên gồm: họ tên, ngày sinh, ngày tuyển dụng, giới tính và lương. Viết chương trình thực hiện các chức năng sau:

- Nhập thông tin cho 5 nhân viên
- In thông tin các nhân viên ra màn hình, mỗi người trên 1 dòng.

- c) Cho biết nhân viên có mức lương cao nhất, thấp nhất.
- d) Cho biết số nhân viên nữ trong danh sách.
- e) Cho biết tổng lương và lương trung bình của các nhân viên.
- f) Công ty tăng lương 20% cho toàn thể nhân viên. Hãy cập nhật.

Câu 8. ☒ Sử dụng con trỏ làm lại Câu 7 - Bài thực hành số 3 và Câu 1 - Bài thực hành số 4.

Câu 9. Cho dãy số gồm n số nguyên ngẫu nhiên. Cài đặt thuật toán xây dựng cây nhị phân tìm kiếm bằng cách lần lượt đưa các nút có giá trị là phần tử dãy số.

VIII. BÀI THỰC HÀNH SỐ 6

Kỹ thuật xử lý tập tin

1. Mục tiêu

Sử dụng thành thạo các thao tác trên tập tin:

- Quy trình thao tác trên tập tin
- Các thao tác xuất nhập dữ liệu
- Các hàm xử lý tập tin

2. Yêu cầu (trước khi đến LAB)

- Nắm vững kiến thức về quy trình sử dụng tập tin
- Làm ra giấy các bài tập mục 3.

3. Các bài tập

Câu 1. ☒ Viết chương trình nhập một dãy n số nguyên a_1, a_2, \dots, a_n . Ghi dãy trên vào tập tin rồi đọc, tính tổng và in kết quả ra màn hình.

Câu 2. ☒ Viết chương trình nhập ghi vào tập tin n phần tử đầu tiên của dãy Fibonacci.

Câu 3. Viết chương trình nhập dữ liệu và ghi vào file một dãy số nguyên bất kỳ. Sắp xếp dãy theo thứ tự tăng dần, rồi lại ghi vào đĩa.

Câu 4. Viết đoạn chương trình để nhập dữ liệu cho file dagiac.inp với cách thức:

- Dòng đầu tiên: Số lượng đỉnh
- Các dòng tiếp theo, mỗi dòng lưu tọa độ một đỉnh
- Hãy in ra màn hình tọa độ các đỉnh của đa giác trên.

Câu 5. ☒ Viết chương trình nhập tạo một tập tin có nội dung là một ma trận vuông n dòng và n cột gồm các phần tử là các số nguyên

- a) nhập từ bàn phím.
- b) ngẫu nhiên không phân biệt từ 1 đến n^2 .

Câu 6. Tập tin INP.DAT có nội dung là một ma trận A vuông cấp n gồm các số nguyên nhập từ bàn phím. Đọc và ghi vào tập tin OUT.DAT là ma trận chuyển vị của A.

Câu 7. Giả sử file A.DAT chứa ma trận A kích thước $m \times n$ các số thực được nhập từ bàn phím. Ghi vào file B.DAT ma trận B cùng kích thước có các phần tử thỏa mãn:

$$b_{ij} = \begin{cases} 1 & \text{khi } a_{ij} > 0 \\ 0 & \text{khi } a_{ij} = 0 \\ -1 & \text{khi } a_{ij} < 0 \end{cases}$$

Câu 8. ☒ Viết chương trình tạo tập tin MATRAN.DAT có nội dung là ma trận kích thước $m \times n$ trong đó các phần tử bằng 0 nếu có tổng các chỉ số là chẵn, các phần tử còn lại bằng 1.

Câu 9. Viết chương trình đếm số chữ cái, số từ, số dòng của một tập tin văn bản.

Câu 10. Cho một tập tin văn bản bất kỳ.

- a) Đếm số từ có chiều dài lớn hơn 4.
- b) Đếm số lần xuất hiện từ X trong tập tin trên, X nhập từ bàn phím.

Câu 11. Viết chương trình copy nội dung một tập tin văn bản sang tập tin khác.

Câu 12. ☒ Viết chương trình tạo file văn bản có tên là "MATRIX.INP" có cấu trúc như sau:

- Dòng đầu ghi hai số m, n.
- Trong m dòng tiếp theo mỗi dòng ghi n số và các số cách nhau một khoảng cách.
Hãy kiểm tra xem trong file đó có bao nhiêu số nguyên tố.
Kết quả cần ghi vào file "MATRIX.OUT" có nội dung là một số nguyên đó là số lượng các số nguyên tố trong file "MATRIX.INP".

Câu 13. Cho số nguyên n, hãy in tam giác PASCAL gồm n dòng.

- Dữ liệu vào: tập tin văn bản PAS.INP gồm 1 dòng chứa giá trị n.
- Kết quả: đưa ra tập tin văn bản PAS.OUT thể hiện một tam giác PASCAL n dòng.

Câu 14. Cho mảng các số nguyên, hãy sắp xếp mảng theo thứ tự tăng dần.

Dữ liệu vào : tập tin văn bản ARRAY.INP gồm 2 dòng

- Dòng 1 chứa số nguyên n ($n \leq 100$).
- Dòng 2 chứa n số nguyên.

Kết quả : Đưa ra tập tin văn bản ARRAY.OUT gồm hai dòng

- Dòng 1 chứa n phần tử của mảng các số nguyên.
- Dòng 2 chứa n số nguyên được xếp tăng dần.

Câu 15. ☒ Cho mảng các số nguyên, tìm phần tử lớn nhất của mảng.

- Dữ liệu vào: tập tin văn bản ARRAY.INP gồm hai dòng:
- Dòng 1 chứa số nguyên n ($n \leq 100$).
- Dòng 2 chứa n số nguyên.

Kết quả: Đưa ra tập tin văn bản ARRAY.OUT gồm 1 dòng ghi 2 giá trị x, y trong đó x là giá trị lớn nhất, y là vị trí của x trong mảng.

Câu 16. Cho mảng các số nguyên, tính tổng các phần tử của mảng.

Dữ liệu vào : tập tin văn bản ARRAY.INP gồm hai dòng

- Dòng 1 chứa số nguyên n ($n \leq 10$)
- Dòng 2 chứa n số nguyên

Kết quả : Đưa ra tập tin văn bản ARRAY.OUT gồm một dòng ghi tổng các phần tử trong mảng.

Câu 17. Cho mảng các số nguyên, hãy liệt kê các phần tử là số nguyên tố

Dữ liệu vào : tập tin văn bản NT.INP gồm hai dòng

- Dòng 1 chứa số nguyên n ($n \leq 100$)
- Dòng 2 chứa n số nguyên

Kết quả : đưa ra tập tin văn bản NT.OUT gồm hai dòng:

- Dòng 1 chứa số lượng các phần tử nguyên tố trong mảng.
- Dòng 2 liệt kê các số nguyên tố đó.

Câu 18. Viết chương trình nhập và lưu hồ sơ của sinh viên vào một file có tên là "DSSV.TXT". Sau đó đọc file "DSSV.TXT" và lưu vào mảng, hãy sắp xếp các hồ sơ sinh viên theo thứ tự giảm dần theo điểm trung bình môn học rồi in ra màn hình hồ sơ các sinh viên theo thứ tự đó ra màn hình có thông tin như sau :

- Mã số sinh viên.
- Họ và tên sinh viên.
- Điểm trung bình kiểm tra.
- Điểm thi hết môn.
- Điểm trung bình môn học (tính bằng (điểm TBKT+điểm thi)/2).

Câu 19. Tạo một file text có tên là “INPUT.TXT” có cấu trúc như sau :

- Dòng đầu tiên ghi hai số M và N (M,N là hai số nguyên dương nhập từ bàn phím).
- Trong M dòng tiếp theo mỗi dòng ghi N số nguyên ngẫu nhiên trong phạm vi từ 0 đến 100 (các số này cách nhau ít nhất một khoảng trắng).

Hãy đọc dữ liệu từ file trên và lưu vào mảng hai chiều. Rồi thực hiện các công việc sau:

- Tìm giá trị lớn nhất của ma trận.
- Đếm số lượng số chẵn, lẻ, nguyên tố có trong ma trận.
- Hãy tính tổng các phần tử trên mỗi dòng của ma trận.
- Hãy ghi kết quả này vào filetext có tên là “OUTPUT.TXT”

Câu 20. Cho dãy (aN), hãy tìm đoạn con tăng dần có tổng lớn nhất

Dữ liệu : được cho trong tập tin AMAX.INP

- Dòng 1 chứa số nguyên N ($0 < N \leq 30000$).
- N dòng tiếp theo, mỗi dòng chứa một số là các số của dãy dãy cho theo đúng thứ tự. Giá trị tuyệt đối của mỗi số không vượt quá 30000.

Kết quả tìm được ghi vào tập tin văn bản AMAX.OUT gồm hai dòng:

- Dòng 1 ghi tổng của dãy con.
- Dòng 2 ghi mảng con tăng dần có tổng lớn nhất.

Câu 21. Viết chương trình nhập lý lịch một nhân viên vào danh sách các nhân viên. Khi không nhập nữa bấm phím Esc và ghi vào tập tin NHANVIEN.DAT sau đó :

- Đọc từ tập tin NHANVIEN.DAT vừa tạo và in danh sách các nhân viên lên màn hình.
- Tìm và in lý lịch một nhân viên bằng các nhập và họ tên hoặc mã số nhân viên.

IX. BÀI THỰC HÀNH SỐ 7

Kỹ thuật lập trình đệ quy

1. Mục tiêu

Sử dụng kỹ thuật lập trình đệ quy, bao gồm:

- Hiểu rõ cơ chế hoạt động của hàm đệ quy
- Cách xây dựng hàm đệ quy
- Các phương pháp khử đệ quy

2. Yêu cầu (trước khi đến LAB)

- Đọc hiểu về phần lý thuyết.
- Làm ra giấy các bài tập mục 3.

3. Các bài tập

Câu 1. ☒ Xây dựng hàm đệ quy cho các 9-11 của Bài thực hành số 2

Câu 2. ☒ Xây dựng hàm đệ quy cho các Câu 1-5 của Bài thực hành số 3

Câu 3. Cài đặt hoàn chỉnh các ví dụ và bài tập phần lý thuyết

Câu 4. ☒ Nhập một mảng n số nguyên. Sử dụng đệ quy tính tổng giá trị các phần tử có trong mảng.

Câu 5. ☒ Nhập 2 số nguyên dương a và b. Sử dụng đệ quy tính ước số chung lớn nhất của 2 số đó.

Câu 6. Nhập số nguyên dương N. Sử dụng đệ quy tính in dãy nhị phân của số N đó.

Câu 7. ☒ Nhập một mảng gồm số N số nguyên. Sử dụng đệ quy kiểm tra xem có phải là mảng tăng dần.

Câu 8. ☒ Nhập một mảng gồm số N số nguyên. Sử dụng đệ quy kiểm tra xem có phải là mảng đối xứng.

Câu 9. ☒ Cho n số tự nhiên x_1, x_2, \dots, x_n . Hãy tìm UCLN (x_1, x_2, \dots, x_n) bằng cách sử dụng: $\text{UCLN}(x_1, x_2, \dots, x_n) = \text{UCLN}(\text{UCLN}(x_1, x_2, \dots, x_{n-1}), x_n)$

Câu 10. Sử dụng đệ quy tìm chữ số có giá trị lớn nhất của số nguyên dương n.

Câu 11. Điền vào những dấu chấm hỏi một trong các phép toán +, -, *, / xem có xảy ra đẳng thức không:

$$(1?(2?(3?(4?(5?(6?(7?(8?9)))))))=35$$

Câu 12. Cho dãy số gồm n số nguyên ngẫu nhiên.

- a) Cài đặt thuật toán xây dựng cây nhị phân tìm kiếm bằng cách lần lượt đưa các nút có giá trị là phần tử dãy số.
- b) Đếm số nút trên cây
- c) Đếm số nút lá, nút có 1 con và nút có 2 con
- d) Đếm số nút mà giá trị của nó là số nguyên tố.