



KỸ THUẬT LẬP TRÌNH

Bài tập 5 – Tuần 13

Kiểu con trỏ, cấp phát động và đệ quy

Phần 1: Bài tập bắt buộc

Bài 1:

Viết chương trình sao chép dữ liệu từ một mảng nguồn sang một mảng đích sao cho mảng nguồn là do người dùng nhập vào và mảng đích chỉ chứa các số chẵn. Việc truy xuất và sao chép dữ liệu giữa hai mảng phải được thực hiện bằng con trỏ.

Ví dụ:

Ban đầu mảng nguồn sẽ do người dùng nhập vào, mảng đích chưa được sao chép dữ liệu:

Mảng nguồn:

7	9	6	1	10	5	2	3	4	8
---	---	---	---	----	---	---	---	---	---

Mảng đích:

--	--	--	--	--	--	--	--	--	--

Sau khi được sao chép dữ liệu:

Mảng nguồn:

7	9	6	1	10	5	2	3	4	8
---	---	---	---	----	---	---	---	---	---

Mảng đích:

6	10	2	4	8					
---	----	---	---	---	--	--	--	--	--

Bài 2:

Sử dụng con trỏ, viết chương trình nghịch đảo một chuỗi.

Bài 3:

Cho đoạn chương trình sau:

```
#include<iostream.h>
#include <stdlib.h>
#include<time.h>

void output(int** c, int m, int n) {
    for (int i = 0; i < m; i++) {
        for (int j = 0; j < n; j++) {
            cout<<c[i][j]<<" ";
        }
        cout<<endl;
    }
}
```



```
    }  
}  
  
void deleteMatrix(int** d, int m) {  
    for (int i = 0; i < m; i++) {  
        delete [] d[i];  
    }  
    delete []d;  
}  
  
void main() {  
    int m, n;  
    int** a;  
    int** b;  
  
    cout<<"Input the dimensions of the Matrix : ";  
    cin>>m>>n;  
    if (m <= 0 || n <= 0) {  
        cout<<"Invalid Input !"<<endl;  
    } else {  
        init(a, m, n);  
        cout<<"The random matrix : "<<endl;  
        output(a, m, n);  
        transposeMatrix(a, m, n, b);  
        cout<<"The transposing matrix : "<<endl;  
        output(b, n, m);  
        deleteMatrix(a, m);  
        deleteMatrix(b, n);  
    }  
}
```

Hãy hiện thực hàm *init()* và hàm *transposeMatrix()* thỏa mãn các yêu cầu sau :

- *init()*: tạo ra ma trận **a** (**m** hàng , **n** cột) chứa các số nguyên bằng cách cấp phát động cho biến con trỏ **a**. Sau đó khởi tạo ma trận **a** chứa các số nguyên **ngẫu nhiên** từ **0..9** (Gợi ý : sử dụng hàm **rand()** trong thư viện **stdlib.h** đã được include sẵn).
- *transposeMatrix()*: tạo ma trận **b** (**n** hàng, **m** cột) chứa các số nguyên bằng cách cấp phát động cho biến con trỏ **b**. Sau đó, tạo ra ma trận **chuyển vị** của ma trận **a** và chứa vào ma trận **b**.

Bài 4:

Viết hàm đệ quy để tính tổng các số nguyên tố từ 2 đến n, với n được cho bởi user. Lưu ý: nếu n không phải là số nguyên tố thì sẽ tính tổng các số nguyên tố từ 2 đến số nguyên tố lớn nhất mà nhỏ hơn n.

Ví dụ: n = 8 => sum = 2 + 3 + 5 + 7



Bài 5:

Viết chương trình tìm giá trị nhỏ nhất của một mảng số nguyên bằng đệ quy.

//return the maximum element in a[]

int findmin(int a[], int n)

Bài 6:

Viết hàm đệ quy sumEvenDigit(int n) thực hiện chức năng sau :

- Trả về -1 nếu $n < 0$.
- Trả về tổng của các chữ số lẻ của số n nếu $n \geq 0$.

Ví dụ : $n = 9568310$ thì kết quả là : $9 + 5 + 3 + 1 = 18$.

Bài 7:

Hiện thực thuật giải tìm ước số chung lớn nhất bằng kỹ thuật đệ quy.

Phần 2: Bài tập làm thêm

Bài 8:

Cho đoạn chương trình :

```
#include <iostream.h>

void printCombinations(int m, int n) {
    //Coding here
}

void main()
{
    int m,n;
    cout<<"Input two positive integer : ";
    cin>>m>>n;
    if (m <=0 || n <= 0 || m > n) {
        cout<<"Invalid input !"<<endl;
    } else {
        cout<<"The result is : "<<endl;
        printCombinations(m,n);
    }
}
```

Hãy hiện thực hàm *printCombinations()* để sinh ra tất cả các tổ hợp chập *m* của *n* phần tử từ *1..n*. Yêu cầu trong hàm *printCombinations()* phải gọi *1 hàm đệ qui* để thực hiện việc sinh ra tất cả các tổ hợp này.

Ví dụ : $m = 2$, $n = 4$. Kết quả xuất ra màn hình sẽ là :

```
1 2
1 3
```



1 4
2 3
2 4
3 4

Bài 9:

Bằng cách sử dụng kỹ thuật hàm đệ quy để viết chương trình kiểm tra xem đảo ngược một chuỗi có phải là chính nó hay không.

//returns 1 if a[] is a palindrome, 0 otherwise

int ispalindrome(char a[], int n)

Bài 10:

Sử dụng hàm đệ quy để viết chương trình đảo ngược một số nguyên.

// with the parameter 12345, it would return 54321

int reverseDigits(int n)

Bài 11:

Viết chương trình giải bài toán tháp Hà Nội.

-- Hết --