

Nội dung tuần 4:

Họ và tên: Nguyễn Huy Đạt

MSSV: 2305CT2334

Bài 1: Viết chương trình cho phép người nhập ma trận mxn theo 2 cách theo bài giảng.

- Họ và tên: Nguyễn Huy Đạt

- MSSV: 2305CT2334

Bài 1

```
#include <iostream>
```

```
using namespace std;
```

```
void chuyendoiMang (int R, int C)
```

```
{
```

```
    int a[R][C];
```

```
    float *** M = new float *[R];
```


// cách 1:

```
cout << "Nhập giá trị cho mảng 2 chiều: " << endl;
```

```
for (int i = 0; i < R; i++)
```

```
{
```

```
    for (int j = 0; j < C; j++)
```

```
    {
```

```
        cout << "a[" << i << "][" << j << "] = ";
```

```
        cin >> a[i][j];
```

```
    }
```

```
    cout << endl;
```

```
}
```

```
cout << "Ma tran mang 2 chieu: " << endl;
```

```
for (int i = 0; i < R; i++)
```

```
{
```

```
    for (int j = 0; j < C; j++)
```

```
    {
```

```
        cout << a[i][j] << " ";
```

```
    }
```

```
    cout << endl;
```

```
}
```

```
int b[R * C];
```

```
for (int i = 0; i < R; i++)
```

```
{
```

```
    for (int j = 0; j < C; j++)
```

```
    {
```

```
        b[i * C + j] = a[i][j];
```

```
    }
```



```
cout << "Ma tran mang 2 chieu su dung **M: " << endl;
for (int i=0; i<R; i++)
{
```

```
    for (int j=0; j<C; j++)
    {
        cout << M[i][j] << " ";
    }
```

```
    cout << endl;
}
```

```
int M[R * C];
for (int i=0; i<R; i++)
{
    for (int j=0; j<C; j++)
    {
        M[i * C + j] = M[i][j];
    }
}
```

```
cout << "*****" << endl;
```

```
cout << "cach 2 " << endl;
```

```
cout << "Chuyen mang 2 chieu thanh mang 1 chieu su dung  
**M: " << endl;
```

```
for (int i=0; i<R * C; i++)
{
    cout << M[i] << " ";
}
```

```
cout << endl;
```

```

        delete[] M,
    }

    int main()
    {
        int m, n;
        cout << "Xin moi nhap hang: ";
        cin >> m;
        cout << "Xin moi nhap cot: ";
        cin >> n;

        chuyen doi Mang (m, n);

        return 0;
    }

```

Bài 2: Dùng đa năng hoá toán tử (overloading) để định nghĩa các phép toán cộng, trừ, nhân chia.

```

    Bài 2
    #include <iostream>
    using namespace std;

    struct so phuc
    {
        double so thuc;
        double so ao;
    };

```



```

void DisplaySP (sophuc Ketqua)
{
    cout << "So thuc: " << Ketqua.sothuc << endl;
    cout << "So phuc: " << Ketqua.soao << endl;
}

```

```

sophuc SetSP (double R, double I)
{
    sophuc Ketqua;
    Ketqua.sothuc = R;
    Ketqua.soao = I;
    return Ketqua;
}

```

```

sophuc operator + (sophuc a, sophuc b)
{
    sophuc Ketqua;
    Ketqua.sothuc = a.sothuc + b.sothuc;
    Ketqua.soao = a.soao + b.soao;
    return Ketqua;
}

```

```

sophuc operator - (sophuc a, sophuc b)
{
    sophuc Ketqua;
    Ketqua.sothuc = a.sothuc - b.sothuc;
    Ketqua.soao = a.soao - b.soao;
    return Ketqua;
}

```


Thời
Ngày

No.

```
sophuc operator * (sophuc a, sophuc b)
```

```
{
```

```
    Sophuc ketqua;
```

```
    ketqua.sothuc = a.sothuc * b.sothuc - a.soao * b.soao;
```

```
    ketqua.soao = a.sothuc * b.soao + a.soao * b.sothuc;
```

```
    return ketqua;
```

```
}
```

```
sophuc operator / (sophuc a, sophuc b)
```

```
{
```

```
    Sophuc ketqua;
```

```
    double mau = (b.sothuc * b.sothuc) + (b.soao * b.soao);
```

```
    ketqua.sothuc = (a.sothuc * b.sothuc + a.soao * b.soao) / mau;
```

```
    ketqua.soao = (a.soao * b.sothuc - a.sothuc * b.soao) / mau;
```

```
    return ketqua;
```

```
}
```

```
int main()
```

```
{
```

```
    Sophuc a, b, c, d, e, f;
```

```
    a = SetSP(5.0, -7.0);
```

```
    b = SetSP(4.0, 2.0);
```

```
    c = a + b;
```

```
    d = a - b;
```

```
    e = a * b;
```

```
    f = a / b;
```

```

cout << "ket qua phap cong la: " << endl;
DisplaySP(c);
cout << endl;

cout << "ket qua phap tru la: " << endl;
DisplaySP(d);
cout << endl;

cout << "ket qua phap phap nhan la: " << endl;
DisplaySP(e);
cout << endl;

cout << "ket qua phap chia la: " << endl;
DisplaySP(f);
cout << endl;

```

3

Bài 3: Cho dãy số sau: `int arr[] = { 1, 4, 2, 3, 6, 5, 8, 9, 7 };` viết chương trình

1) sắp xếp dãy số trên từ bé tới lớn.

2) Cho phép người dùng nhập 1 số tìm kiếm bất kì. Nếu có số đó trong dãy trên thì trả về vị trí của số đó. Ngược lại hiện thị kết quả "Không tìm thấy". Sử dụng phương pháp nhị phân.

Bài 3

```

#include <iostream>
using namespace std;

void BubbleSort (int arr [])
{
    for (int i=0; i<9-1; i++)
    {
        for (int j=i+1; j<9; j++)

```



```
{
    if (arr[i] > arr[j])
    {
        int tmp = arr[i];
        arr[i] = arr[j];
        arr[j] = tmp;
    }
}

int BinarySearch(int arr[], int trai, int phai, int so can tim)
{
    while (trai <= phai)
    {
        int mid = trai + (phai - trai) / 2;
        if (arr[mid] == so can tim)
        {
            return mid;
        }
        if (arr[mid] < so can tim)
        {
            trai = mid + 1;
        }
        else
        {
            phai = mid - 1;
        }
    }
}
```



```
    }  
    return -1;  
}
```

```
int main()
```

```
{
```

```
    int arr[9] = {1, 4, 2, 3, 6, 5, 8, 9, 7};
```

```
    int Size = sizeof(arr) / sizeof(arr[0]);
```

```
    int i;
```

```
    BubbleSort(arr);
```

```
    cout << "Mang da duoc sap xep: ";
```

```
    for (int i = 0; i < 9; i++)
```

```
    {
```

```
        cout << arr[i] << " ";
```

```
    }
```

```
    cout << endl;
```

```
    cout << "Moi nhap so de tim kien: ";
```

```
    cin >> i;
```

```
    int ketqua = BinarySearch(arr, 0, Size - 1, i);
```

```
    if (ketqua != -1)
```

```
    {
```

```
        cout << "Vi tri cua so " << i << " ] tai vi tri: "
```

```
        << ketqua << endl;
```

```
    }
```



```
else
```

```
{
```

```
    cout << " không tìm thấy " << endl;
```

```
}
```

```
return 0;
```

```
}
```