Họ và tên: Đỗ Hữu Thiên

MSSV: 625104C071

Lớp: CQ.62.KTCĐT

BÁO CÁO BÀI TẬP LAB2

PHẦN II: Bài tập thực hành về biến con trỏ và cấp phát động trong mảng 1 chiều

Câu 9:

- *Souce code hàm void nhapMang(int*& arr, int& n);*

```
void nhapMang(int* a, int& n) {
      for (int i = 0; i < n; ++i) {
  cout << "Nhap phan tu thu a[" << i + 1 << "]=  ";
  cin >> *(a + i);
}
}
```

- *Source code hàm xuatMang(int* a, int& n);*

```
void xuatMang(int* a, int n) {

    for (int i = 0; i < n; ++i) {
        cout << *(a + i) << "\t ";
    }
    cout << endl;
}
```

- *Source code hàm hoanvi(int* a, int n);*

```
 void hoanvi(int* x,int* y){
    int temp = *x;
    *x=*y;
    *y=temp;
}
```

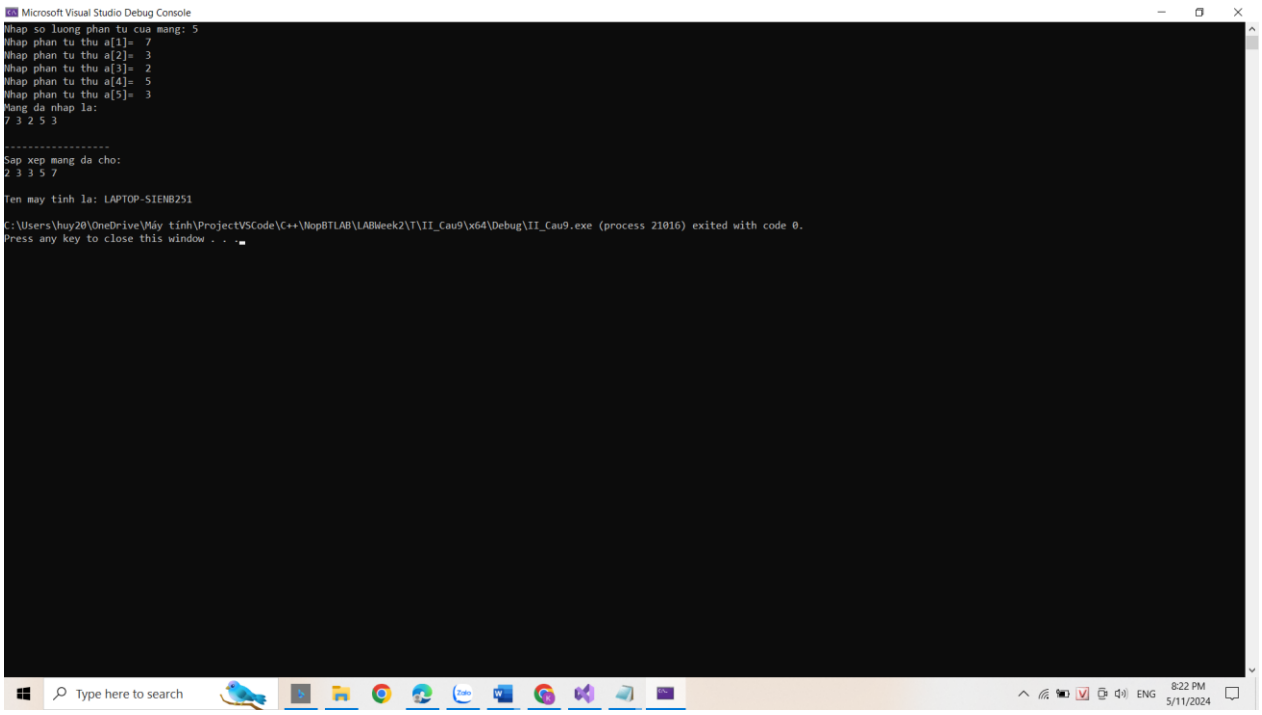
- *Source code hàm SelectionSort(int* a, int n);*

```
 void SelectionSort(int* a,int n){
    int doicho;
    for (int i=0;i<n-1;i++){
        doicho=i;
        for (int j=i+1;j<n;j++)
            if (a[j]<a[doicho])
                doicho = j;
        hoanvi(&a[doicho],&a[i]);
    }
}
```

❖ *Source code hàm main cho Câu 9:*

```
int main(){
    int n;
    cout << "Nhap so luong phan tu cua mang: ";
    cin >> n;
    int* a = new int[n];
    nhapMang(a,n);
    cout << "Mang da nhap la:\n";
    xuatMang(a,n);
    cout << "\n------------------";
    SelectionSort(a, n);
    cout << "\nSap xep mang da cho: " << endl;
    for (int i = 0; i < n; i++) {
        cout << a[i] << " ";
    }
    cout << endl;
    // Sử dụng hàm getenv để lấy tên máy tính từ biến môi trường COMPUTERNAME
    char computerName[MAX_COMPUTERNAME_LENGTH + 1];
    DWORD size = sizeof(computerName);

    if (GetComputerNameA(computerName, &size)) {
        cout << "\nTen may tinh la: " << computerName << endl;
    }
    else {
        cout << "\nKhong the lay ten may tinh." << endl;
    }
    return 0;
}
```

❖ *Hình chụp kết quả*



Câu 10:

- *Souce code hàm void nhapMang(int*& arr, int& n);*

```
void nhapMang(int* a, int& n) {
      for (int i = 0; i < n; ++i) {
  cout << "Nhap phan tu thu a[" << i + 1 << "]=  ";
  cin >> *(a + i);
}
}
```

- *Source code hàm xuatMang(int* a, int& n);*

```
void xuatMang(int* a, int n) {

    for (int i = 0; i < n; ++i) {
        cout << *(a + i) << "\t ";
    }
    cout << endl;
}
```

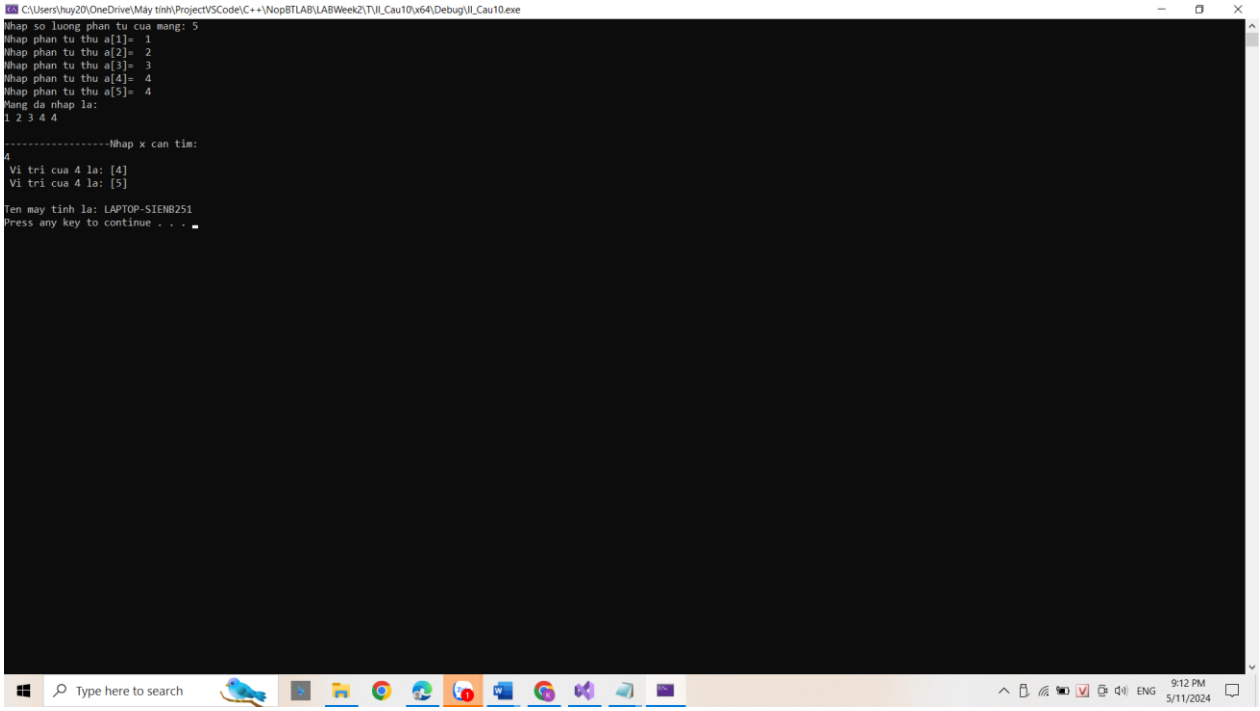
- *Source code hàm timKiem(int *a,int n , int x);*

```
void timKiem(int* a, int n, int x) {
    int kiemtra = 0;
    for (int i = 0; i < n; i++) {
        if (a[i] == x) {
            cout << " Vi tri cua " << x << " la: [" << i+1 << "]" << endl;
            kiemtra = 1;
        }
    }
    if (kiemtra == 0) {
        cout << " Khong tim thay " << x << " trong mang da nhap" << endl;
    }
}
```

- *Source code hàm main.*

```
int main() {
  int n,x;
  cout << "Nhap so luong phan tu cua mang: ";
  cin >> n;
  int* a = new int[n];
  nhapMang(a, n);
  cout << "Mang da nhap la:\n";
  xuatMang(a, n);
  cout << "\n------------------";
  cout << "Nhap x can tim: " << endl;
  cin >> x;
  timKiem(a, n, x);
  delete[] a;
  char computerName[MAX_COMPUTERNAME_LENGTH + 1];
  DWORD size = sizeof(computerName);
  if (GetComputerNameA(computerName, &size)) {
      cout << "\nTen may tinh la: " << computerName << endl;
  }
  else {
      cout << "\nKhong the lay ten may tinh." << endl;
  }
  system("pause");
  return 0;
}
```

❖ Hình chụp kết quả:



Câu 11:

❖ *Souce code hàm void nhapMang(int*& arr, int& n);*

```
void nhapMang(int* a, int& n) {
      for (int i = 0; i < n; ++i) {
  cout << "Nhap phan tu thu a[" << i + 1 << "]=  ";
  cin >> *(a + i);
}
}
```

❖ *Source code hàm xuatMang(int* a, int& n);*

```
void xuatMang(int* a, int n) {

    for (int i = 0; i < n; ++i) {
        cout << *(a + i) << "\t ";
    }
    cout << endl;
}
```

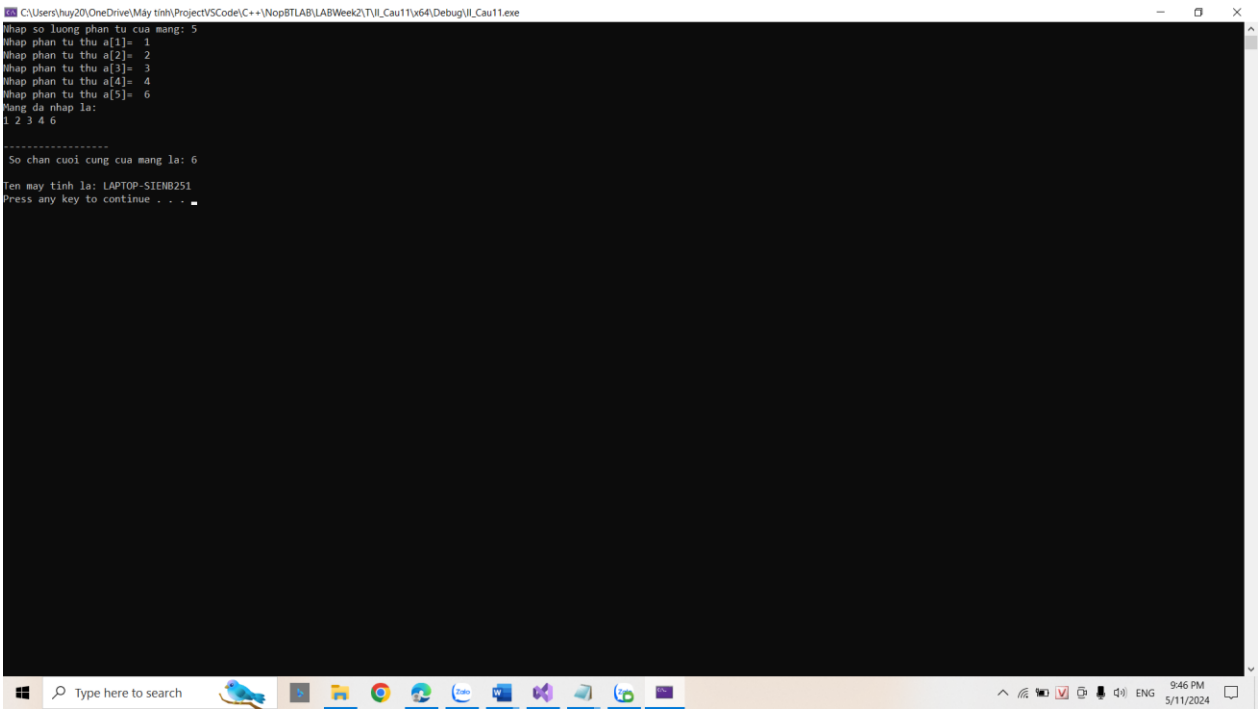
❖ *Source code hàm* int timChanCuoiCung(int* arr, int n);

```
int timChanCuoiCung(int* a, int n) {
    int kiemtra = 0;
    for (int i = 0; i < n; i++) {
        if (a[i] % 2 == 0) {
           kiemtra = a[i];
        }
    }
    if (kiemtra == 0) {
        return 0;
    }
    return kiemtra;
}
```

❖ *Soucre code hàm main();*

```
int main() {
    int n;
    cout << "Nhap so luong phan tu cua mang: ";
    cin >> n;
    int* a = new int[n];
    nhapMang(a, n);
    cout << "Mang da nhap la:\n";
    xuatMang(a, n);
    cout << "\n------------------";
    int kiemtra = timChanCuoiCung(a, n);
    if (kiemtra != 0) {
        cout << "So chan cuoi cung cua mang la: " << kiemtra << endl;
    }
    else {
        cout << "Khong co gia tri chan trong mang." << endl;
    }
    delete[] a;
    char computerName[MAX_COMPUTERNAME_LENGTH + 1];
    DWORD size = sizeof(computerName);
    if (GetComputerNameA(computerName, &size)) {
        cout << "\nTen may tinh la: " << computerName << endl;
    }
    else {
        cout << "\nKhong the lay ten may tinh." << endl;
    }
    system("pause");
    return 0;
}
```

❖ Hình chụp kết quả



Câu 12:

- *Souce code hàm void nhapMang(int*& arr, int& n);*

```
void nhapMang(int* a, int& n) {
      for (int i = 0; i < n; ++i) {
  cout << "Nhap phan tu thu a[" << i + 1 << "]=  ";
  cin >> *(a + i);
}
}
```

- *Source code hàm xuatMang(int* a, int& n);*

```
void xuatMang(int* a, int n) {

    for (int i = 0; i < n; ++i) {
        cout << *(a + i) << "\t ";
    }
    cout << endl;
}
```

- *Source code hàm* `int timsoduong(int* a, int n);`

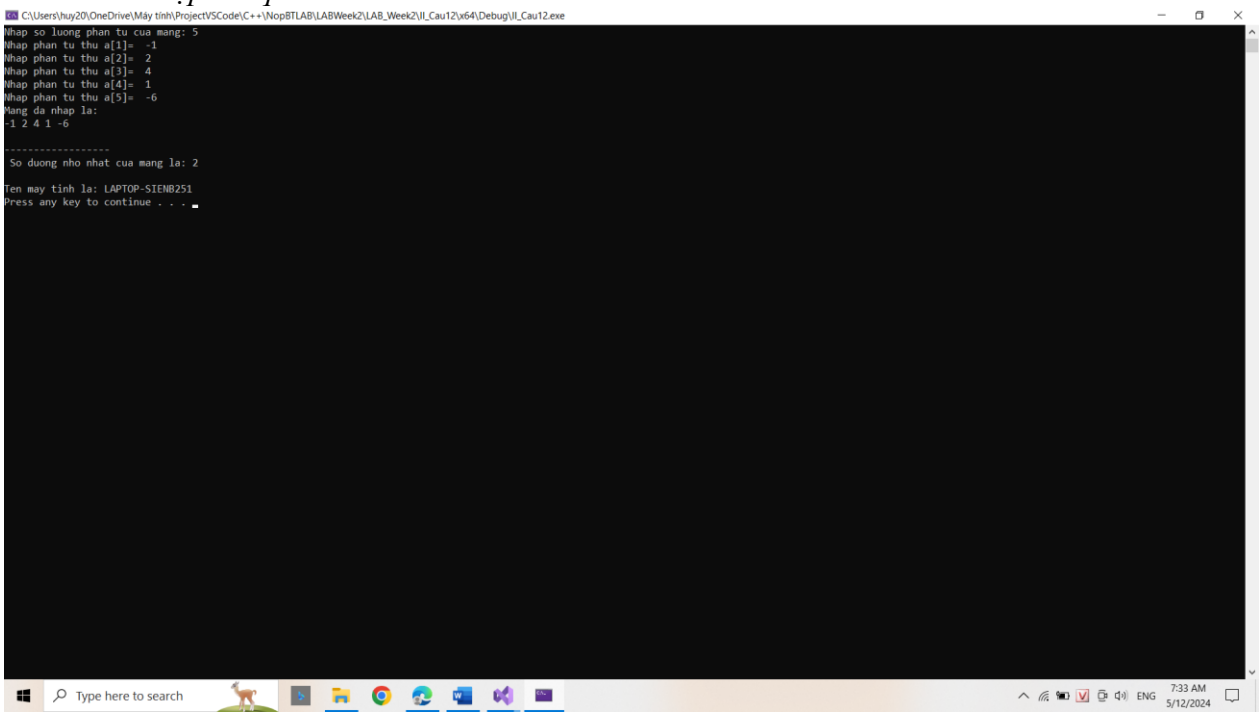
```
int timsoduong(int* a, int n) {
    int min = 0;
    for (int i = 0; i < n; i++) {
        if (a[i] > 0) {
            if (a[i] < a[i + 1]) {
                min = a[i];
            }
        }
    }
    if (min == 0) {
        return -1;
    }
    return min;
}
```

- *Source code hàm* `main();`

```
int main() {
    int n;
    cout << "Nhap so luong phan tu cua mang: ";
    cin >> n;
    int* a = new int[n];
    nhapMang(a, n);
    cout << "Mang da nhap la:\n";
    xuatMang(a, n);
    cout << "\n------------------";
    int kiemtra = timsoduong(a, n);
    if (kiemtra != -1) {
        cout << "\n So duong nho nhat cua mang la: " << kiemtra << endl;
    }
    else {
        cout << "Khong co gia tri duong. Suy ra " << kiemtra <<endl;
    }
    delete[] a;
    char computerName[MAX_COMPUTERNAME_LENGTH + 1];
    DWORD size = sizeof(computerName);
    if (GetComputerNameA(computerName, &size)) {
        cout << "\nTen may tinh la: " << computerName << endl;
    }
    else {
```

```
            cout << "\nKhong the lay ten may tinh." << endl;
        }
        system("pause");
        return 0;
    }
```

❖ *Hình chụp kết quả*



Câu 13:

❖ *Souce code hàm void nhapMang(int*& arr, int& n);*

```
    void nhapMang(int* a, int& n) {
          for (int i = 0; i < n; ++i) {
      cout << "Nhap phan tu thu a[" << i + 1 << "]=  ";
      cin >> *(a + i);
  }
    }
```

❖ *Source code hàm xuatMang(int* a, int& n);*

```
    void xuatMang(int* a, int n) {

        for (int i = 0; i < n; ++i) {
            cout << *(a + i) << "\t ";
        }
        cout << endl;
    }
```

❖ *Source code hàm kiemtrasont(int n);*

```
    bool kiemtrasont(int n) {
        int b[8] = {2,3,4,5,6,7,8,9};
        int kiemtra{};
        if (n < 2) {
            return false;
        }
        if (n > 2 && n < 9) {
            int arr[] = { 2, 3, 5, 7 };
            for (int i = 0; i < 4; i++)
```

```
                if (n == arr[i]) return true;
            return false;
        }
        else if (n > 9) {
            for (int i = 1; i < 9; i++) {
                if (n % b[i] == 0) {
                    kiemtra = false;
                    return false;
                }
            }
        }
        else if (kiemtra != false) {
                return true;
            }
    }
```

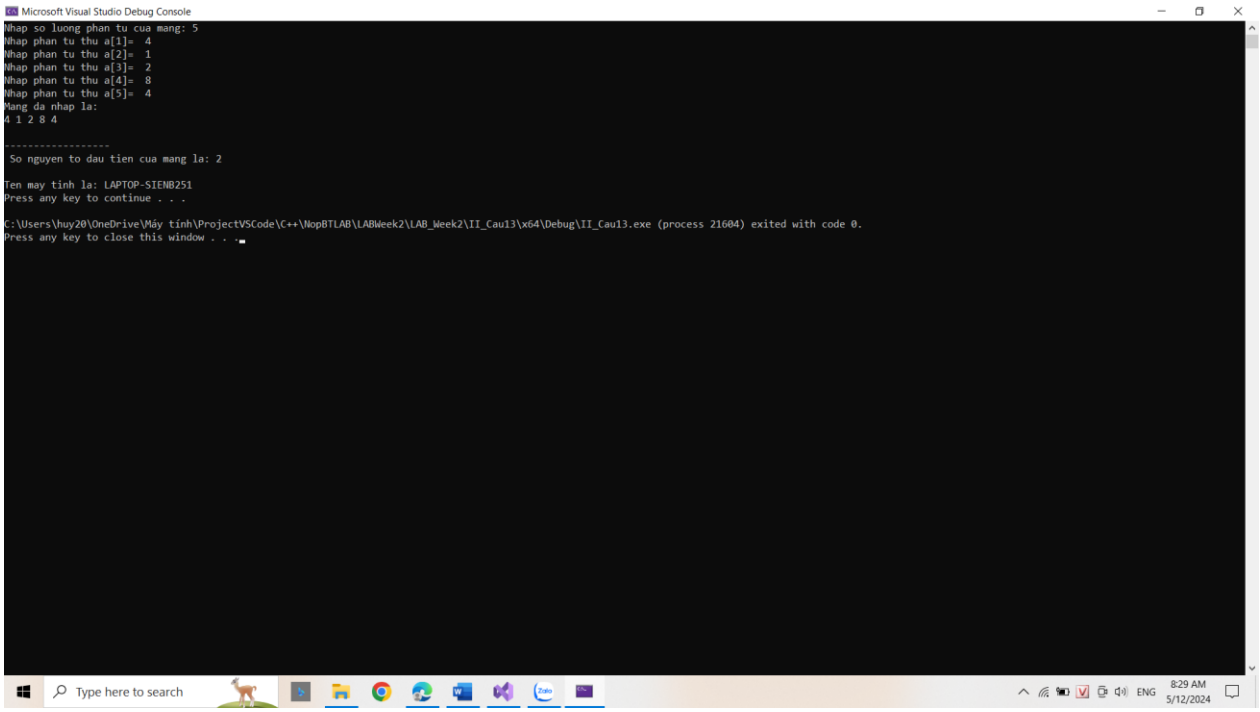
❖ *Source code hàm* `timSoNguyenToDauTien(int* a, int n);`

```
int timSoNguyenToDauTien(int* a, int n) {
    for (int i = 0; i < n; i++) {
        if (kiemtrasont(a[i]) == true) {
           return a[i];
        }
    }
    return -1;
}
```

❖ *Source code hàm main();*

```
int main() {
    int n;
    cout << "Nhap so luong phan tu cua mang: ";
    cin >> n;
    int* a = new int[n];
    nhapMang(a, n);
    cout << "Mang da nhap la:\n";
    xuatMang(a, n);
    cout << "\n------------------";
    int kiemtra = timSoNguyenToDauTien(a, n);
    if (kiemtra != -1) {
        cout << "\n So nguyen to dau tien cua mang la: " << kiemtra << endl;
    }
    else {
        cout << "Khong co so nguyen to trong mang.Suy ra " << kiemtra << endl;
    }
    delete[] a;
    char computerName[MAX_COMPUTERNAME_LENGTH + 1];
    DWORD size = sizeof(computerName);
    if (GetComputerNameA(computerName, &size)) {
        cout << "\nTen may tinh la: " << computerName << endl;
    }
    else {
        cout << "\nKhong the lay ten may tinh." << endl;
    }
    system("pause");
    return 0;
}
```

❖ *Hình chụp kết quả:*



Câu 14:

- *Souce code hàm void nhapMang(int*& arr, int& n);*

```
void nhapMang(int* a, int& n) {
      for (int i = 0; i < n; ++i) {
  cout << "Nhap phan tu thu a[" << i + 1 << "]=  ";
  cin >> *(a + i);
}
}
```

- *Source code hàm xuatMang(int* a, int& n);*

```
void xuatMang(int* a, int n) {

    for (int i = 0; i < n; ++i) {
        cout << *(a + i) << "\t ";
    }
    cout << endl;
}
```

- *Source code hàm int timPhanTuItNhat(int* arr, int n);*

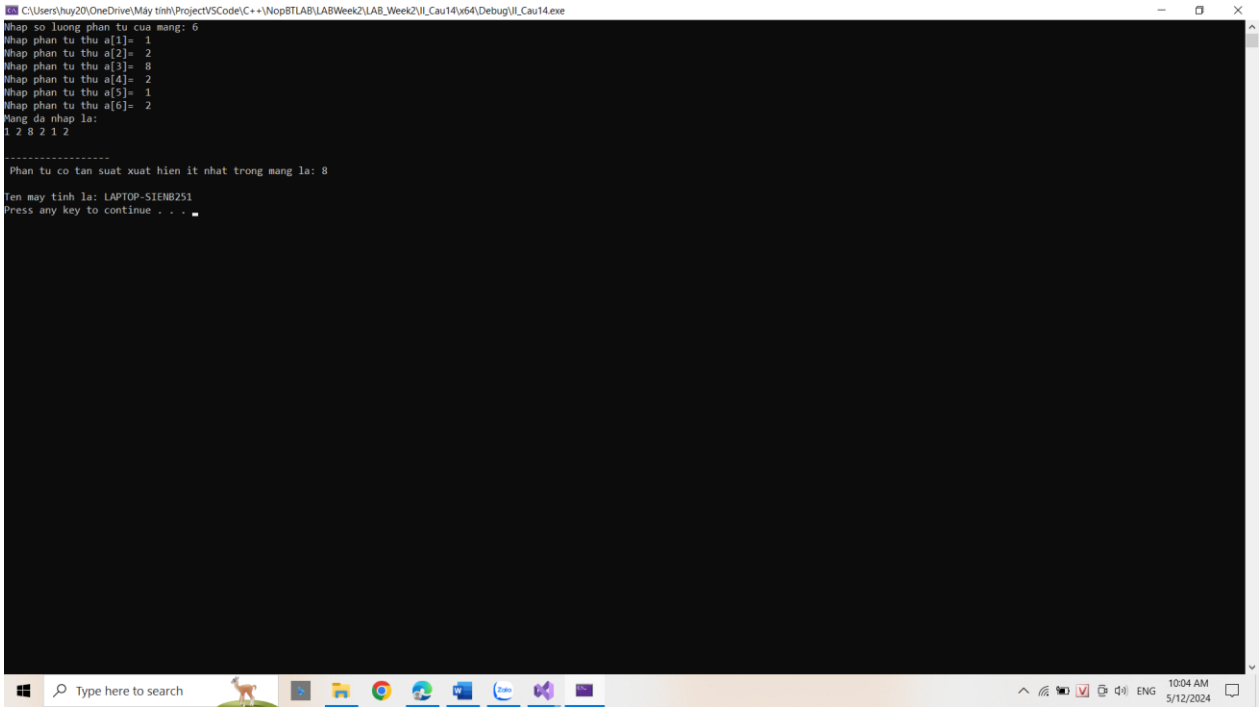
```
int timPhanTuItNhat(int* arr, int n) {
    int phantucantim = arr[0];
    int tansuat = n;
    for (int i = 1; i < n; i++) {
        int soLanXuatHien = 0;
        for (int j = 0; j < n; j++) {
            if (arr[i] == arr[j]) {
                soLanXuatHien++;
            }
        }
        if (soLanXuatHien < tansuat) {
            phantucantim = arr[i];
            tansuat = soLanXuatHien;
        }
```

```
        }
        return phantucantim;
    }
```

❖ *Source code hàm main();*

```
int main() {
    int n;
    cout << "Nhap so luong phan tu cua mang: ";
    cin >> n;
    int* a = new int[n];
    nhapMang(a,n);
    cout << "Mang da nhap la:\n";
    xuatMang(a,n);
    cout << "\n------------------";
    int phanTuItNhat = timPhanTuItNhat(a,n);
    cout << "\n Phan tu co tan suat xuat hien it nhat trong mang la: " <<
phanTuItNhat << endl;
    delete[] a;
    char computerName[MAX_COMPUTERNAME_LENGTH + 1];
    DWORD size = sizeof(computerName);
    if (GetComputerNameA(computerName, &size)) {
        cout << "\nTen may tinh la: " << computerName << endl;
    }
    else {
        cout << "\nKhong the lay ten may tinh." << endl;
    }
    system("pause");
    return 0;
    }
```

❖ Hình chụp kết quả:



PHẦN III: Bài tập thực hành về lớp và đối tượng, nạp chồng toán tử trong C++

Câu 2: Viết chương trình nhập dữ liệu vào từ bàn phím để tạo hai đối tượng số phức x, y. Tính và in ra tổng, hiệu 2 số phức đó.

❖ *Source code Complex.h*

```
#pragma once
#include<iostream>
using namespace std;
class Complex
{
private:
    double pt;
    double pa;
public:
    Complex();
    Complex(double p_thuc, double p_ao);
    void NhapSoPhuc();
    void InSoPhuc();
    Complex operator+(const Complex& x);
};
```

❖ *Source code Complex.cpp*

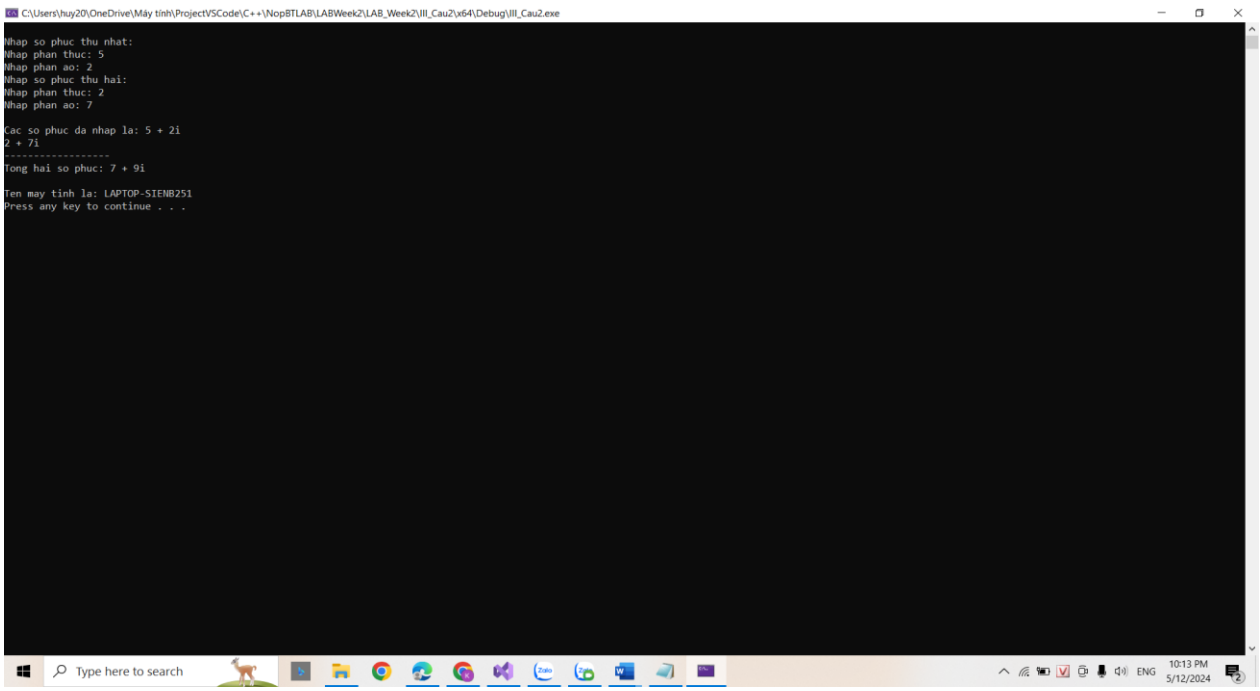
```
#include "Complex.h"
Complex::Complex()
{
 pt = 0.0;
 pa = 0.0;
}

Complex::Complex(double p_thuc, double p_ao)
{
 pt = p_thuc;
 pa = p_ao;
}

void Complex::NhapSoPhuc()
{
 cout << "Nhap phan thuc: ";
 cin >> pt;
 cout << "Nhap phan ao: ";
 cin >> pa;
}

void Complex::InSoPhuc()
{
 if (pa > 0)
       cout << pt << " + " << pa << "i";
 else cout << pt << pa << "i";
}

Complex Complex::operator+(const Complex& x)
{
 Complex temp;
 temp.pt = this->pt + x.pt;
 temp.pa = this->pa + x.pa;
```

```
    return temp;

}
```

❖ *Source code file Main.cpp*

```
#include<iostream>
#include<cstdlib>
#include<windows.h>
using namespace std;
#include "Complex.h"
int main() {
    Complex x, y;
    cout << "\nNhap so phuc thu nhat:\n";
    x.NhapSoPhuc();
    cout << "Nhap so phuc thu hai:\n";
    y.NhapSoPhuc();
    cout << "\nCac so phuc da nhap la: ";
    x.InSoPhuc();
    cout << endl;
    y.InSoPhuc();
    cout << "\n------------------";
    cout << "\nTong hai so phuc: ";
    Complex tong = x + y;
    tong.InSoPhuc();
    cout << endl;
    char computerName[MAX_COMPUTERNAME_LENGTH + 1];
    DWORD size = sizeof(computerName);

    if (GetComputerNameA(computerName, &size)) {
        cout << "\nTen may tinh la: " << computerName << endl;
    }
    else {
        cout << "\nKhong the lay ten may tinh." << endl;
    }
    system("pause");
    return 0;
}
```

❖ **Hình chụp kết quả**



Câu 4:

❖ *Source code PS.h*

```
#pragma once
#include <iostream>
using namespace std;
class PS{
private:
    int tu, mau;
public:
    PS();
    PS(int x, int y);
    void Nhap();
    void Xuat();
    int UCLN(int x, int y);
    void RutGon();
    PS operator+(PS);
    PS operator-(PS);
    PS operator*(PS);
    PS operator/(PS);
    bool operator>(PS);
};
```

❖ *Source code PS.cpp*

```
#include<iostream>
#include "PS.h"
using namespace std;
PS::PS() {
    tu = 0;
    mau = 1;
}
PS::PS(int a, int b){
    tu = a;
    mau = b;
}
void PS::Nhap(){
    cout << "Nhap tu so: ";
    cin >> tu;
    cout << "Nhap mau so: ";
    cin >> mau;
}
void PS::Xuat(){
    cout << tu << "/" << mau;
}
int PS::UCLN(int x, int y)
{
    for (int i = min(x, y); i > 0; i--) {
        if (x % i == 0 && y % i == 0)
            return i;
    }
}
void PS::RutGon(){
    int ucln = UCLN(tu, mau);
    tu /= ucln;
    mau /= ucln;
}
PS PS::operator+(PS a){
```

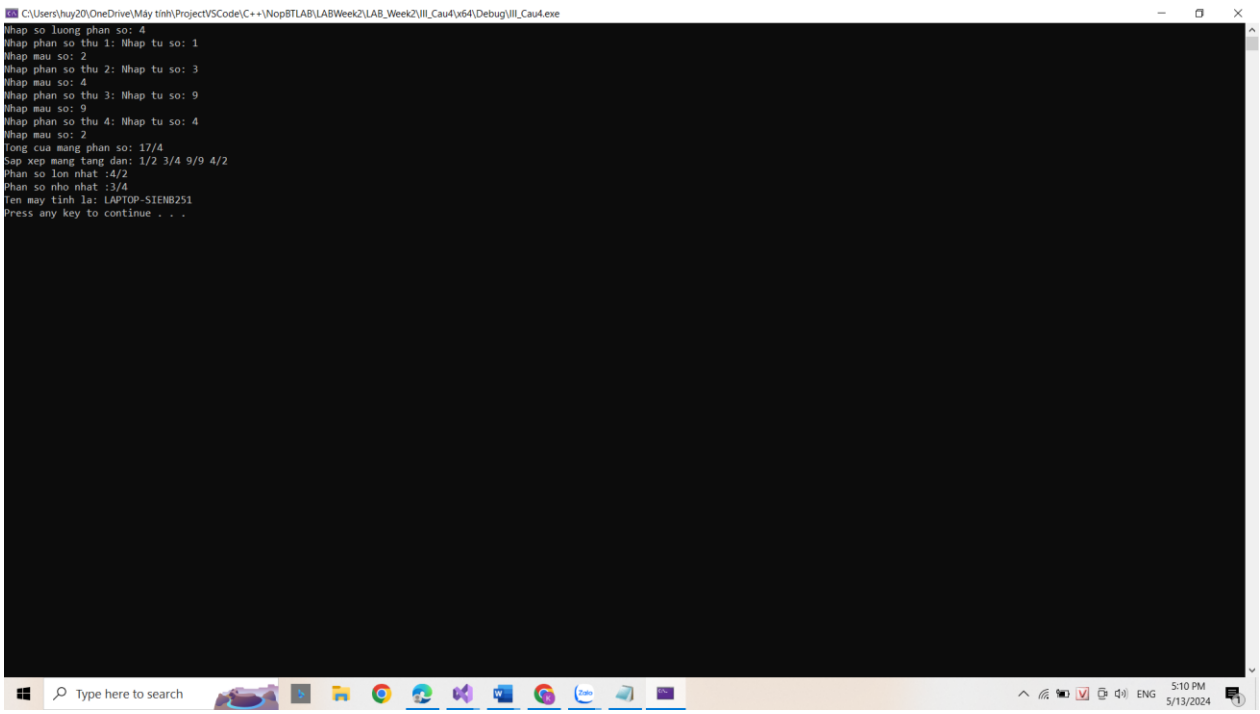
```
    PS tong;
    tong.tu = tu * a.mau + mau * a.tu;
    tong.mau = mau * a.mau;
    tong.RutGon();
    return tong;
}
PS PS::operator-(PS a){
    PS hieu;
    hieu.tu = tu * a.mau - mau * a.tu;
    hieu.mau = mau * a.mau;
    hieu.RutGon();
    return hieu;
}
PS PS::operator*(PS a){
    PS tich;
    tich.tu = tu * a.tu;
    tich.mau = mau * a.mau;
    tich.RutGon();
    return tich;
}
PS PS::operator/(PS a){
    PS thuong;
    thuong.tu = tu * a.mau;
    thuong.mau = mau * a.tu;
    thuong.RutGon();
    return thuong;
}
bool PS::operator>(PS a){
    return (float)tu / mau < (float)a.tu / a.mau;
}
```

❖ *Source code Main.cpp*

```
#include<iostream>
#include "PS.h"
#include<cstdlib>
#include<windows.h>
using namespace std;
void swap(int* a, int* b){
    int temp = *b;
    *b = *a;
    *a = temp;
}
int main() {
    int n;
    cout << "Nhap so luong phan so: ";
    cin >> n;
    PS* a = new PS[n];
    for (int i = 0; i < n; i++){
        cout << "Nhap phan so thu " << i + 1 << ": ";
        a[i].Nhap();
    }
    PS tong;
    for (int i = 0; i < n; i++){
        tong = tong + a[i];
    }
    cout << "Tong cua mang phan so: ";
    tong.Xuat();
    cout << endl;
    for (int i = 0; i < n - 1; i++){
        for (int j = i + 1; j < n; j++){
```

```
                if (a[j] > a[i]) {
                    swap(a[i], a[j]);
                }
            }
        }
        PS min = a[1];
        PS max = a[n-1];
        cout << "Sap xep mang tang dan: ";
        for (int i = 0; i < n; i++){
            a[i].Xuat();
            cout << " ";
        }
        cout << "\nPhan so lon nhat :";
        max.Xuat();
        cout << "\nPhan so nho nhat :";
        min.Xuat();
        delete[] a;
        char computerName[MAX_COMPUTERNAME_LENGTH + 1];
        DWORD size = sizeof(computerName);

        if (GetComputerNameA(computerName, &size)) {
            cout << "\nTen may tinh la: " << computerName << endl;
        }
        else {
            cout << "\nKhong the lay ten may tinh." << endl;
        }
        system ("Pause");
        return 0;
}
```

❖ *Hình chụp kết quả*



Câu 5:

- *Source code Student.h*

```
#pragma once
#include<iostream>
#include <format>
#include <string>
using namespace std;
class Student {
    private:
        string ten;
        double diemky1, diemky2;
    public:
        Student();
        Student(string, double, double);
        string laytenSV();
        void Nhap();
        void Xuat();
        bool KiemtraDTB();
};
```

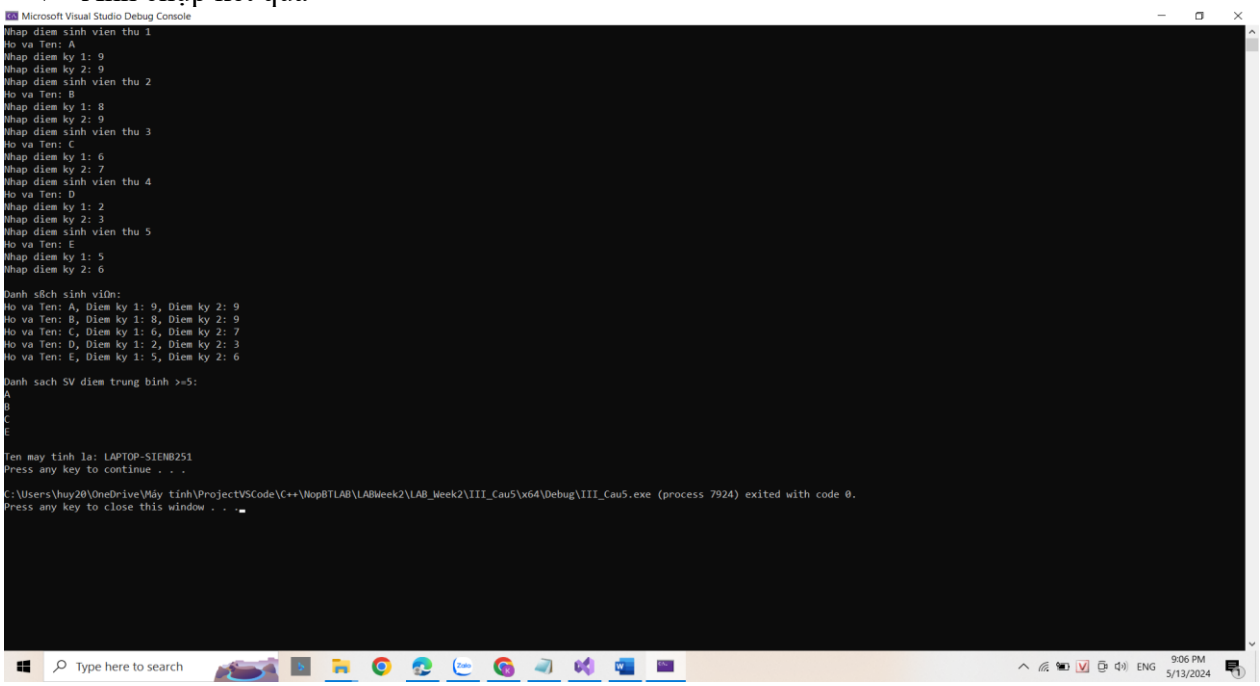
- *Source code Student.cpp*

```
#include<iostream>
#include <format>
#include <string>
using namespace std;
#include "Student.h"
Student::Student() {
    ten = "";
    diemky1 = diemky2 = 0.0;
}

Student::Student(string n, double s1, double s2) {
    ten = n;
    diemky1 = s1;
    diemky2 = s2;
}

string Student::laytenSV()
{
    return ten;
}

void Student::Nhap() {
    cout << "Ho va Ten: ";
    getline(cin, ten);
    cout << "Nhap diem ky 1: ";
    cin >> diemky1;
    cout << "Nhap diem ky 2: ";
    cin >> diemky2;
}

void Student::Xuat() {
    cout << "Ho va Ten: " << ten << ", Diem ky 1: " << diemky1<< ", Diem ky
2: " << diemky2 << endl;
}
bool Student::KiemtraDTB() {
    return ((diemky1 + 2 * diemky2) / 3) >= 5;
}
```

- *Source code Main.cpp*

```cpp
#include<iostream>
#include <format>
#include <string>
using namespace std;
#include "Student.h"
#include<cstdlib>
#include<windows.h>
int main() {
    Student students[5];
    for (int i = 0; i < 5; i++) {
        cout << "Nhap diem sinh vien thu " << i + 1 << endl;
        students[i].Nhap();
        cin.ignore();
    }
    cout << "\nDanh sách sinh viên:\n";
    for (int i = 0; i < 5; i++) {
        students[i].Xuat();
    }
    cout << "\nDanh sach SV diem trung binh >=5:\n";
    for (int i = 0; i < 5; i++) {
        if (students[i].KiemtraDTB()) {
            cout << students[i].laytenSV() << endl;
        }
    }
    delete[] students;
    char computerName[MAX_COMPUTERNAME_LENGTH + 1];
    DWORD size = sizeof(computerName);
    if (GetComputerNameA(computerName, &size)) {
        cout << "\nTen may tinh la: " << computerName << endl;
    }
    else {
        cout << "\nKhong the lay ten may tinh." << endl;
    }
    system("Pause ");
    return 0;
}
```

- Ảnh chụp kết quả



Câu 8:

- *Source code Date.h*

```
#pragma once
class Date
{
    private:
    int Day, Month, Year;
    public:
        Date();
        void NhapDate();
        void XuatDate();
};
```

- *Source code Date.cpp*

```
#include "Date.h"
#include<iostream>
using namespace std;

Date::Date()
{
    int Day(0);
    int Month(0);
    int Year(0);
}

void Date::NhapDate()
{
    cout << "Nhap ngay: ";
    cin >> Day;
    cout << "Nhap thang: ";
    cin >> Month;
    cout << "Nhap nam: ";
    cin >> Year;
}

void Date::XuatDate()
{
    cout << Day << "/" << Month << "/" << Year;
}
```

- *Source code Nhansu.h*

```
#pragma once
#include<iostream>
#include <string>
#include "Date.h"
using namespace std;
class Nhansu
{
private:
    string Manhansu, Hoten;
    Date NS;
public:
    Nhansu();
    void NhapNS();
    void XuatNS();
};
```

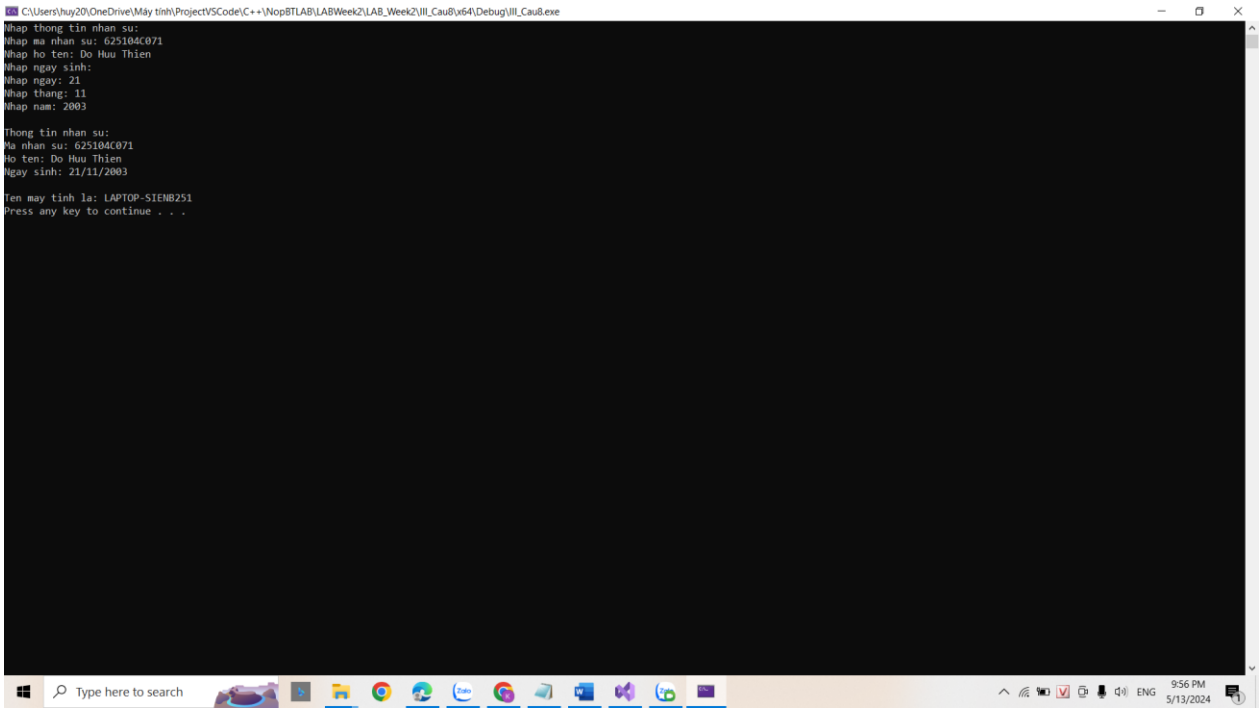
- *Source code Nhansu.cpp*

```cpp
#include "Nhansu.h"
#include<iostream>
#include<string>
using namespace std;

Nhansu::Nhansu()
{
	string Nhansu();
	string Manhansu("");
	string Hoten("");
}
void Nhansu::NhapNS()
{
	cout << "Nhap ma nhan su: ";
	getline(cin, Manhansu);
	cout << "Nhap ho ten: ";
	getline(cin, Hoten);
	cout << "Nhap ngay sinh:\n";
	NS.NhapDate();
}
void Nhansu::XuatNS()
{
	cout << "Ma nhan su: " << Manhansu << endl;
	cout << "Ho ten: " << Hoten << endl;
	cout << "Ngay sinh: "; NS.XuatDate(); cout << endl;
}
```

- *Source code Main.cpp*

```cpp
#include "Nhansu.h"
#include <iostream>
using namespace std;
#include<cstdlib>
#include<windows.h>
int main() {
    Nhansu x;
    cout << "Nhap thong tin nhan su:\n";
    x.NhapNS();
    cout << "\nThong tin nhan su:\n";
    x.XuatNS();
    char computerName[MAX_COMPUTERNAME_LENGTH + 1];
    DWORD size = sizeof(computerName);
    if (GetComputerNameA(computerName, &size)) {
        cout << "\nTen may tinh la: " << computerName << endl;
    }
    else {
        cout << "\nKhong the lay ten may tinh." << endl;
    }
    system("pause");
    return 0;
}
```

- *Hình Chụp kết quả*



PHẦN IV: Bài tập thực hành tính kế thừa của OOP trong C++\

Câu 2:

- *Source code MyDate .h*

```
#pragma once
#include <iostream>

using namespace std;
class MyDate {
    int day, month, year;
public:
    MyDate();
    MyDate(int day, int month, int year);
    void Nhapdate();
    void Indate();
    bool operator > (MyDate& date);
};
```

- *Source code MyDate.cpp*

```
#include "MyDate.h"
MyDate::MyDate(){}
MyDate::MyDate(int day, int month, int
year):day(day),month(month),year(year){}
void MyDate::Nhapdate()
{
     cout << "\nNhap (day month year): ";
     cin >> day >> month >> year;
```

```
}
void MyDate::Indate()
{
	cout << day<< "/" <<month << "/" <<year << endl;
}
bool MyDate::operator>(MyDate& date)
{
	if (year != date.year)
		return year > date.year;
	if (month != date.month)
		return month > date.month;
	return day > date.day;
}
```

❖ *Source code Person.h*

```
#pragma once
#include "MyDate.h"
class Person : public MyDate {
private:
    string name, address, phone;
public:
    Person();
    Person(string name, string address, string phone, int d, int m, int y);
    void NhapTT();
    void XuatTT();
};
```

❖ *Source code Person.cpp*

```
#include "Person.h"
#include "MyDate.h"
#include <string>
#include<iostream>
using namespace std;
Person::Person(){}
Person::Person(string name, string address, string phone, int d, int m, int
y)
	:name(name),address(address),phone(phone),MyDate(d,m,y) {}
void Person::NhapTT()
{
	cout << "\nNhap ten nhan vien: ";
	cin >> name;
	cout << "\nNhap dia chi tin nhan vien: ";
	cin >> address;
	cout << "\nNhap SDT nhan vien: ";
	cin >> phone;
	MyDate::Nhapdate();
}
void Person::XuatTT()
{
	cout << "Name: " << name << ", \nAddress: " << address << ", \nPhone: "
<< phone << ", \nBirthdate: ";
	MyDate::Indate();
}
```

❖ *Source code Officer .h*

```
#pragma once
#include"Person.h"
#include "MyDate.h"
#include <string>
#include<iostream>
using namespace std;
class Officer : public Person {
private:
    float salary;
public:
    Officer();
    Officer(string name, string address, string phone, int d, int m, int y,
float s);
    void Nhapofficer();
    void Xuatofficer();
};
```

❖ *Source code Officer.cpp*

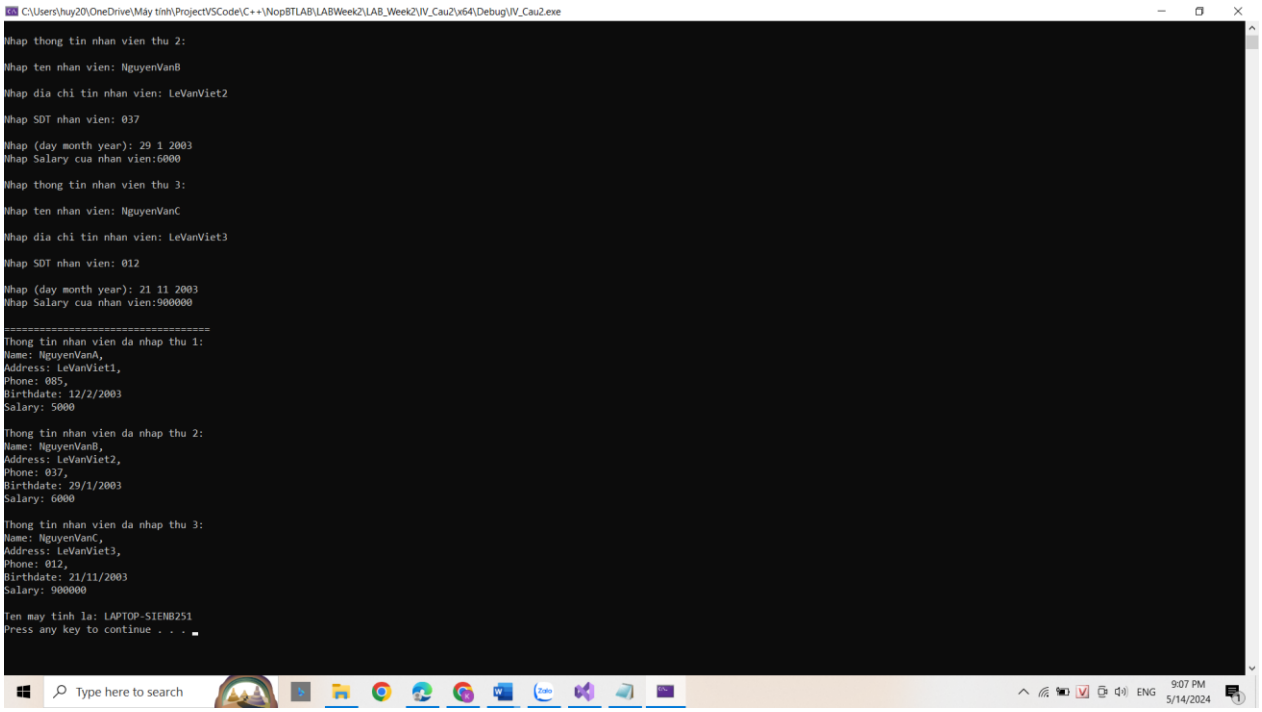
```
#include "Officer.h"
Officer::Officer() : salary(0.0) {}
Officer::Officer(string n, string a, string p, int d, int m, int y, float
salary)
     :Person(n,a,p,d,m,y),salary(salary) {}
void Officer::Nhapofficer()
{
     Person::NhapTT();
     cout << "Nhap Salary cua nhan vien:";
     cin >> salary;
}
void Officer::Xuatofficer()
{
     Person::XuatTT();
     cout << "Salary: " << salary << endl;
}
```

❖ *Source code Main.cpp*

```
#include "Officer.h"
#include <iostream>
using namespace std;
#include<cstdlib>
#include<windows.h>
int main() {
    int n;
    cout << "Nhap so luong nhan vien: ";
    cin >> n;
    cout << "\n------------";
    Officer* dsOfficer = new Officer[n];
    for (int i = 0; i < n; i++) {
        cout << "\nNhap thong tin nhan vien thu " << i + 1 << ":\n";
        dsOfficer[i].Nhapofficer();
    }
    cout << "\n==================================";
    for (int i = 0; i < n; i++) {
        cout << "\nThong tin nhan vien da nhap thu " << i + 1 << ":\n";
        dsOfficer[i].Xuatofficer();
```

```
        }
        delete[] dsOfficer;
        char computerName[MAX_COMPUTERNAME_LENGTH + 1];
        DWORD size = sizeof(computerName);

        if (GetComputerNameA(computerName, &size)) {
            cout << "\nTen may tinh la: " << computerName << endl;
        }
        else {
            cout << "\nKhong the lay ten may tinh." << endl;
        }
        system("pause");
        return 0;
    }
```

❖ *Ảnh chụp kết quả:*



Câu 7:

❖ *Source code Mytime .h*

```
#pragma once
#include <iostream>
using namespace std;
class Mytime {
protected:
    int hour;
    int min;
    int sec;
public:
    Mytime();
    Mytime(int hr, int mn, int sc);
    void Nhaptime();
    void Xuattime();
};
```

❖ *Source code Mytime .cpp*

```
#include "Mytime.h"

Mytime::Mytime()
{
     hour = 1;
     min = 1;
     sec = 1;
}

Mytime::Mytime(int hr, int mn, int sc)
{
     hour = hr;
     min = mn;
     sec = sc;
}

void Mytime::Nhaptime() {
    do {
        cout << "\nNhap gio: ";
        cin >> hour;
        if (hour < 0 || hour > 24) {
            cout << "\nNgay nhap khong dung! Nhap lai." << endl;
        }
    } while (hour < 0 || hour > 24);

    do {
        cout << "Nhap phut: ";
        cin >> min;
        if (min < 0 || min > 60) {
            cout << "\nThang nhap khong dung. Nhap lai!" << std::endl;
        }
    } while (min < 0 || min > 60);

    do {
        cout << "Nhap giay: ";
        cin >> sec;
        if (sec < 0 || sec > 60) {
            cout << "\nThang nhap khong dung. Nhap lai!" << std::endl;
        }
    } while (sec < 0 || sec > 60);
```

```
}


void Mytime::Xuattime() {
    cout << hour << ":" << min << "-" << sec << endl;
}
```

❖ *Source code Mydate.h*

```
#pragma once
#include<iostream>
using namespace std;
class Mydate {
private:
    int day, month, year;

public:
    Mydate();
    Mydate(int d, int m, int y);
    void NhapDate();
    void XuatDate();
    bool operator>=(const Mydate& other);
};
```

❖ *Source code Mydate.cpp*

```
#include "Mydate.h"
#include <iostream>
using namespace std;
Mydate::Mydate() {
    day = 1;
    month = 1;
    year = 2000;
}

Mydate::Mydate(int d, int m, int y) {
    day = d;
    month = m;
    year = y;
}

void Mydate::NhapDate() {
    do {
        cout << "\nNhap ngay: ";
        cin >> day;
        if (day < 1 || day > 31) {
            cout << "\nNgay nhap khong dung! Nhap lai." << endl;
        }
    } while (day < 1 || day > 31);

    do {
        cout << "Nhap thang: ";
        cin >> month;
        if (month < 1 || month > 12) {
            cout << "\nThang nhap khong dung. Nhap lai!" << std::endl;
        }
    } while (month < 1 || month > 12);

    cout << "\nNhap nam: ";
    cin >> year;
}
```

```
void Mydate::XuatDate() {
    cout << day << "-" << month << "-" << year << endl;
}

bool Mydate::operator>=(const Mydate& other) {
    if (year > other.year)
        return true;
    else if (year == other.year && month > other.month)
        return true;
    else if (year == other.year && month == other.month && day >= other.day)
        return true;
    return false;
}
```

❖ *Source code Myfile .h*

```
#pragma once
#include "Mytime.h"
#include "Mydate.h"
#include <string>
using namespace std;

class Myfile : public Mytime, public Mydate {
private:
    string filename;
    int filesize;

public:
    Myfile();
    Myfile(int hr, int mn, int sc, int d, int m, int y, string name, int
size);
    void XuatThongTinFile();
    void NhapThongTinFile();
    bool operator > (const Myfile& x);
    void Xuatfilemax();
};
```

❖ *Source code Myfile .cpp*

```
#include "Myfile.h"
#include <iostream>

using namespace std;
Myfile::Myfile() {
    Mytime();
    Mydate();
    filename = "";
    filesize = 0;
}

Myfile::Myfile(int hr, int mn, int sc, int d, int m, int y, string name, int
size)
    : Mytime(hr,mn,sc), Mydate(d, m, y), filename(name), filesize(size) {}

void Myfile::NhapThongTinFile() {
    Nhaptime();
    NhapDate();
    cout << "Nhap ten file: ";
    cin >> filename;
    cout << "Nhap kich thuoc file: ";
```

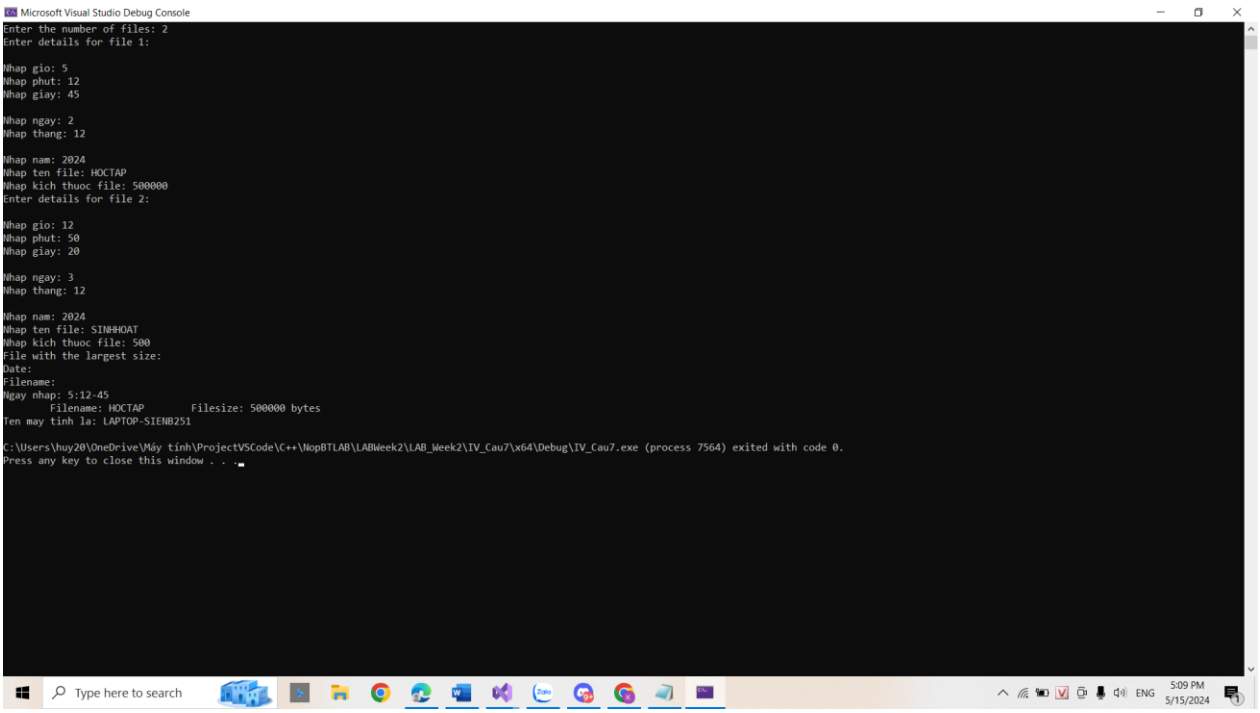
```
    cin >> filesize;
}

bool Myfile::operator>(const Myfile& x)
{
    return filesize > x.filesize;
}

void Myfile::XuatThongTinFile() {
    Xuattime();
    XuatDate();
    cout << "Filename: " << filename << endl;
    cout << "Filesize: " << filesize << " bytes" << endl;
}
void Myfile::Xuatfilemax() {
    cout << "\nNgay nhap: ";
    Xuattime();
    cout << "\tFilename: " << filename << "\tFilesize: " << filesize << "
bytes";
}
```

❖ *Source code Main.cpp*

```
#include <iostream>
#include "Myfile.h"
#include<cstdlib>
#include<windows.h>
using namespace std;
int main() {
    int n;
    cout << "Enter the number of files: ";
    cin >> n;

    Myfile* files = new Myfile[n];

    for (int i = 0; i < n; ++i) {
        cout << "Enter details for file " << i + 1 << ":\n";
        files[i].NhapThongTinFile();
    }

    Myfile largestFile = files[0];
    for (int i = 1; i < n; ++i) {
        if (files[i] > largestFile) {
            largestFile = files[i];
        }
    }
    cout << "File with the largest size:\n";
    cout << "Date: ";
    cout << "\nFilename: ";
    largestFile.Xuatfilemax();
    delete[] files;
    // Sử dụng hàm getenv để lấy tên máy tính từ biến môi trường
COMPUTERNAME
    char computerName[MAX_COMPUTERNAME_LENGTH + 1];
    DWORD size = sizeof(computerName);

    if (GetComputerNameA(computerName, &size)) {
        cout << "\nTen may tinh la: " << computerName << endl;
    }
    else {
        cout << "\nKhong the lay ten may tinh." << endl;
    }
```

```
        return 0;
    }
```

❖ *Chụp hình kết quả:*



Câu 10:

❖ *Source code Mycolor .h*

```
#pragma once
#include<iostream>
#include<string>
using namespace std;
class MyColor {
private:
      string sohieumau;
public:
      MyColor();
      MyColor(string sh);
      void Nhapsohieumau(string sh);
      void Xuatsohieumau();
};
```

❖ *Source code Mycolor.cpp*

```
#include<iostream>
#include"MyColor.h"
using namespace std;
MyColor::MyColor() {}
MyColor::MyColor(string sh) : sohieumau(sh) {}
void MyColor::Nhapsohieumau(string sh) {
      cout << "Nhap so hieu mau: ";
      cin.ignore();
      getline(cin, sohieumau);
}
```

```
void MyColor::Xuatsohieumau() {
	cout << "So hieu mau: " << sohieumau << endl;
}
```

❖ *Source code Point .h*

```
#pragma once
#include<iostream>
class Point {
private:
	int x;
	int y;
public:
	Point();
	Point(int xx, int yy);
	void thietlap(int xx, int yy);
	void Nhaptoado();
	void Xuattoado();
};
```

❖ *Source code Point.cpp*

```
#include<iostream>
#include"Point.h"
using namespace std;
Point::Point() {
	x = 0;
	y = 0;
}
Point::Point(int xx, int yy) {
	x = xx;
	y = yy;
}
void Point::thietlap(int xx, int yy) {
	x = xx;
	y = yy;
}
void Point::Nhaptoado() {
	cout << "\nNhap toa do x: ";
	cin >> x;
	cout << "\nNhap toa do y: ";
	cin >> y;
}
void Point::Xuattoado() {
	cout << "  (" << x << "," << y << ") " << endl;
}
```

❖ *Source code Circle .h*

```
#pragma once
#include<iostream>
#include<cmath>
#include"MyColor.h"
#include"Point.h"
class Circle :public MyColor {
private:
	Point toado;
	float r;
public:
	Circle() :MyColor(), toado(), r(0) {}
	Circle(string sh, int x, int y, float r) :MyColor(sh), toado(x, y),
r(r) {}
	float S();
	void thietlap(string sh, int x, int y, float r);
```

```
        void XuatCircle();
        bool operator > (Circle& a);
};
```

❖ *Source code Circle.cpp*

```
#include<iostream>
#include<cmath>
#include"Circle.h"
using namespace std;
double pi = 3.14;
void Circle::thietlap(string sh, int xx, int yy, float rr) {
    MyColor::Nhapsohieumau(sh);
    toado.thietlap(xx, yy);
    r = rr;
}
float Circle::S() {
    return pi * r * r;
}
void Circle::XuatCircle() {
    Xuatsohieumau();
    cout << "\nDiem: ";
    toado.Xuattoado();
    cout << "\nBan kinh: " << r;
    cout << "\nDien tich: " << S() << endl;
}
bool Circle::operator>(Circle& c) {
    return this->S() > c.S();
}
```

❖ *Source code Main.cpp*

```
#include<iostream>
#include"Circle.h"
using namespace std;
#include<cstdlib>
#include<windows.h>
int main() {
    int n;
    cout << "\nNhap so luong duong tron:  ";
    cin >> n;
    Circle* Cir = new Circle[n];
    for (int i = 0; i < n; i++) {
        int x, y;
        string shmau;
        float bankinh;
        cout << "\nNhap mau cua duong tron " << i + 1 << ": ";
        cin >> shmau;
        cout << "\nNhap toa do x cua duong tron " << i + 1 << ": ";
        cin >> x;
        cout << "\nNhap toa do y cua duong tron " << i + 1 << ": ";
        cin >> y;
        cout << "\nNhap ban kinh cua duong tron " << i + 1 << ": ";
        cin >> bankinh;
        Cir[i].thietlap(shmau, x, y, bankinh);
    }
    Circle max = Cir[0];
    for (int i = 1; i < n; i++) {
        if (Cir[i] > max) {
            max = Cir[i];
        }
    }
    cout << "\nDuong tron co ban kinh lon nhat: " << endl;
```

```
        max.XuatCircle();
        delete[] Cir;
        // Sử dụng hàm getenv để lấy tên máy tính từ biến môi trường
    COMPUTERNAME
        char computerName[MAX_COMPUTERNAME_LENGTH + 1];
        DWORD size = sizeof(computerName);

        if (GetComputerNameA(computerName, &size)) {
            cout << "\nTen may tinh la: " << computerName << endl;
        }
        else {
            cout << "\nKhong the lay ten may tinh." << endl;
        }
        system("pause");
        return 0;
    }
```

❖ *Hình chụp kết quả*