IT002.O216 - 23520161 - BT2

1. PhanSo

a. main.cpp

```cpp
#include <iostream>

#include "phanso.h"
using namespace std;

int main() {
  PhanSo a, b;
  a.Nhap();
  b.Nhap();
  cout << "a + b = ";
  (a + b).Rutgon().Xuat();
  cout << "a - b = ";
  (a - b).Rutgon().Xuat();
  cout << "a * b = ";
  (a * b).Rutgon().Xuat();
  cout << "a / b = ";
  (a / b).Rutgon().Xuat();
  return 0;
}
```

b. phanso.h

```cpp
#pragma once
class PhanSo
{
  public:
    PhanSo();
    ~PhanSo();
    void Nhap();
    void Xuat() const;
    PhanSo Rutgon();
    PhanSo operator+(PhanSo &) const;
    PhanSo operator-(PhanSo &) const;
    PhanSo operator*(PhanSo &) const;
    PhanSo operator/(PhanSo &) const;

  private:
    int tu;
    int mau;
};
```

c. phanso.cpp

```cpp
#include "phanso.h"

#include <cmath>
#include <iostream>
using namespace std;

PhanSo::PhanSo() {
  this->tu = 0;
  this->mau = 1;
}

PhanSo::~PhanSo() {}

void PhanSo::Nhap() {
  do {
    cout << "Nhap tu so: ";
    cin >> this->tu;
    cout << "Nhap mau so: ";
    cin >> this->mau;
  } while (this->mau == 0);
}

void PhanSo::Xuat() const {
  if (this->mau == 1)
    cout << this->tu;
  else if (this->mau == -1)
    cout << this->tu;
  else
    cout << this->tu << "/" << this->mau;
  cout << endl;
}

int maxUoc(int a, int b) {
  int m = abs(a);
  int n = abs(b);
  while (m * n != 0) {
    if (m > n)
      m -= n;
    else
      n -= m;
  }
  return m + n;
}

PhanSo PhanSo::Rutgon() {
  PhanSo res;
  int i = maxUoc(this->mau, this->tu);
  if (this->mau < 0) {
    res.tu = this->tu / (-i);
    res.mau = this->mau / (-i);
```

```cpp
  } else {
    res.tu = this->tu / i;
    res.mau = this->mau / i;
  }
  return res;
}

PhanSo PhanSo::operator+(PhanSo &b) const {
  PhanSo c;
  c.tu = this->tu * b.mau + this->mau * b.tu;
  c.mau = this->mau * b.mau;
  return c;
}

PhanSo PhanSo ::operator*(PhanSo &b) const {
  PhanSo c;
  c.tu = this->tu * b.tu;
  c.mau = this->mau * b.mau;
  return c;
}

PhanSo PhanSo::operator-(PhanSo &b) const {
  PhanSo c;
  c.tu = this->tu * b.mau - this->mau * b.tu;
  c.mau = this->mau * b.mau;
  return c;
}

PhanSo PhanSo::operator/(PhanSo &b) const {
  PhanSo c;
  c.tu = this->tu * b.mau;
  c.mau = this->mau * b.tu;
  return c;
}
```

2. Candidate

a. main.cpp

```cpp
#include <iostream>
#include <vector>

#include "Candidate.h"

using namespace std;

int main() {
  vector<Candidate> Candidates;
  int n;
  cout << "Nhap so luong thi sinh: ";
  cin >> n;
```

```cpp
  Candidates.resize(n);
  for (int i = 0; i < n; i++) {
    cout << "Nhap thong tin thi sinh thu " << i + 1 << endl;
    Candidates[i].Nhap();
  }
  cout << "Cac thi sinh co diem tong lon hon 15: " << endl;
  for (int i = 0; i < n; i++)
    if (Candidates[i].DiemHon15()) Candidates[i].Xuat();
  return 0;
}
```

b. Candidate.h

```cpp
#include <iostream>
#include <vector>

#include "Candidate.h"

using namespace std;

int main() {
  vector<Candidate> Candidates;
  int n;
  cout << "Nhap so luong thi sinh: ";
  cin >> n;
  Candidates.resize(n);
  for (int i = 0; i < n; i++) {
    cout << "Nhap thong tin thi sinh thu " << i + 1 << endl;
    Candidates[i].Nhap();
  }
  cout << "Cac thi sinh co diem tong lon hon 15: " << endl;
  for (int i = 0; i < n; i++)
    if (Candidates[i].DiemHon15()) Candidates[i].Xuat();
  return 0;
}
```

c. Candidate.cpp

```cpp
#include "Candidate.h"

#include <iostream>
#include <string>

using namespace std;

Candidate::Candidate()
    : maSo(""), hoTen(""), diemToan(0), diemVan(0), diemAnh(0), diemTong(0){};

void Candidate::Nhap() {
```

```cpp
  cout << "Nhap ma so sinh vien: ";
  cin.ignore();
  getline(cin, this->maSo);
  cout << "Nhap ho ten sinh vien: ";
  getline(cin, this->hoTen);
  cout << "Nhap diem toan: ";
  cin >> this->diemToan;
  cout << "Nhap diem van: ";
  cin >> this->diemVan;
  cout << "Nhap diem anh: ";
  cin >> this->diemAnh;
  this->TinhDiemTong();
}

bool Candidate::DiemHon15() const { return this->diemTong > 15; }

void Candidate::Xuat() const {
  cout << "Hoc sinh: " << this->hoTen << " - Diem tong: " << this->diemTong
       << endl;
}

void Candidate::TinhDiemTong() {
  this->diemTong = this->diemToan + this->diemVan + this->diemAnh;
}
```