

Test 4 POO

```
#include <iostream>

using namespace std;

class Multiplu
{
    int a,b;

public:
    Multiplu();
    Multiplu(int, int);
    Multiplu(Multiplu &copie);
    ~Multiplu();
    void multiplu();
};

Multiplu::Multiplu()
{
    cout<<"\n\nS-a apelat constructorul vid.";
    cout<<"\nIntroduceti primul numar: ";
    cin>>a;
    cout<<"Introduceti al doilea numar: ";
    cin>>b;
}
```

```
Multiplu::Multiplu(int a, int b)
{
    cout<<"S-a apelat constructorul cu parametrii.\n";
    this->a = a;
    this->b = b;
}
```

```
Multiplu::Multiplu(Multiplu &copie)
{
    cout<<"\nS-a apelat constructorul de copiere.\n";
    a = copie.a;
    b = copie.b;
}
```

```
Multiplu::~~Multiplu()
{
    cout<<"\nS-a apelat destructorul.";
}
```

```
void Multiplu::multiplu()
{

    int cmmmc;

    if(a>b)
        cmmmc = a;
```

```
else
    cmmmc = b;
while(true)
{
    if( cmmmc%a==0 && cmmmc%b==0 )
    {
        cout<<"Numerele "<<a<<" si "<<b<<" au cmmmc "<<cmmmc<<endl;
        break;
    }

    cmmmc++;
}

}

int main()
{
    cout << "Cel mai mic multiplu comun a doua numere" << endl;

    int a,b;
    char tasta_oprire;

    do
    {
        Multiplu m1;
        m1.multiplu();
    }
```

```
cout<<"\nIntroduceti primul numar pentru calculul cmmmc: ";
cin>>a;
cout<<"Introduceti al doilea numar pentru calculul cmmmc: ";
cin>>b;
Multiplu m2(a,b);
m2.multiplu();

Multiplu m3(m2);
m3.multiplu();

cout<<"\nDoriti sa continuati>('d' = DA/ 'n' = NU) ";
cin>>tasta_oprire;

while(tasta_oprire != 'd' && tasta_oprire != 'n')
{
    cout<<"Ai tastat o optiune eronata\n"<<"Tasteaza 'd' pentru DA si 'n' pentru NU: ";
    cin>>tasta_oprire;
}

}while(tasta_oprire != 'n');

return 0;
}
```

