Experiment #6

1.

#include<iostream>

using namespace std;

int myPower(int b, int e) {

if (e == 1)

return b;

else {

e--;

b \*= myPower(b, e);

}

}

int main() {

int base = 0, exponent = 1;

cout << "base = ";

cin >> base;

cout << "exponent = ";

cin >> exponent;

while (exponent <1) {

cout << "\nexponent should be an integer greater than or equal to 1!\n\nexponent = ";

cin >> exponent;

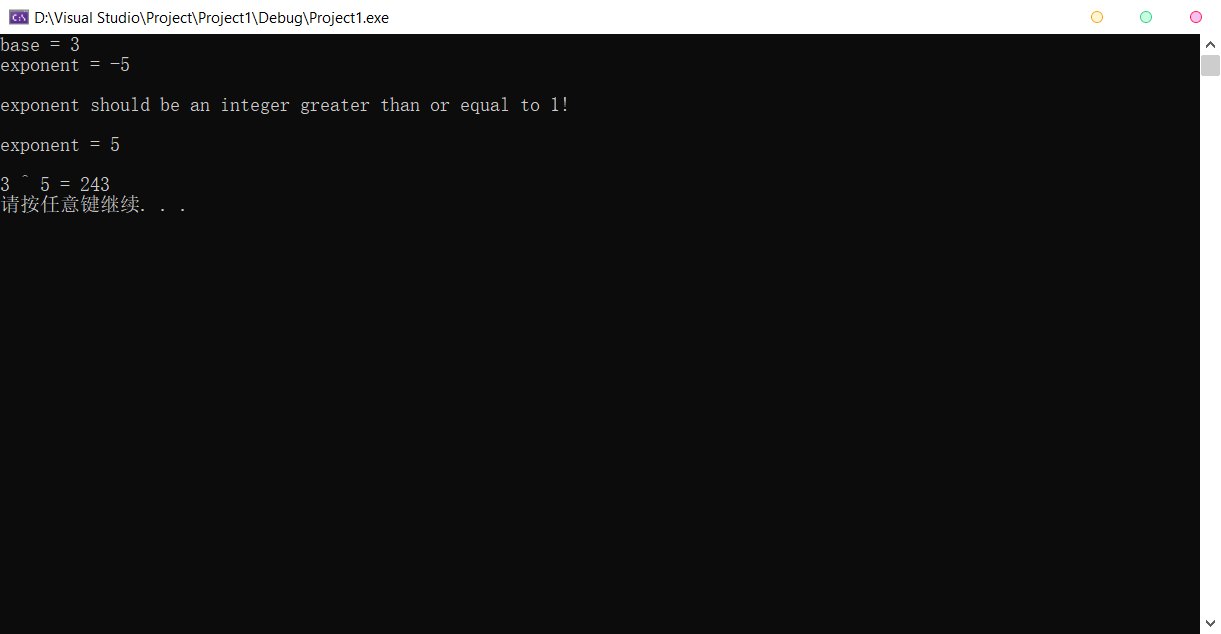
}

cout << endl << base << " ^ " << exponent << " = " << myPower(base, exponent) << endl;

system("pause");

return 0;

}



2 & 3.

#include<iostream>

using namespace std;

int GCD(int num1, int num2) {

if (num1 < num2)

GCD(num2, num1);

if (num1 % num2 == 0)

return num2;

else

return GCD(num2, (num1 % num2));

}

int LCM(int num1, int num2) {

return num1 \* num2 / GCD(num1, num2);

}

int main() {

int num1 = 0, num2 = 0;

cout << "Enter two positive integers.\n";

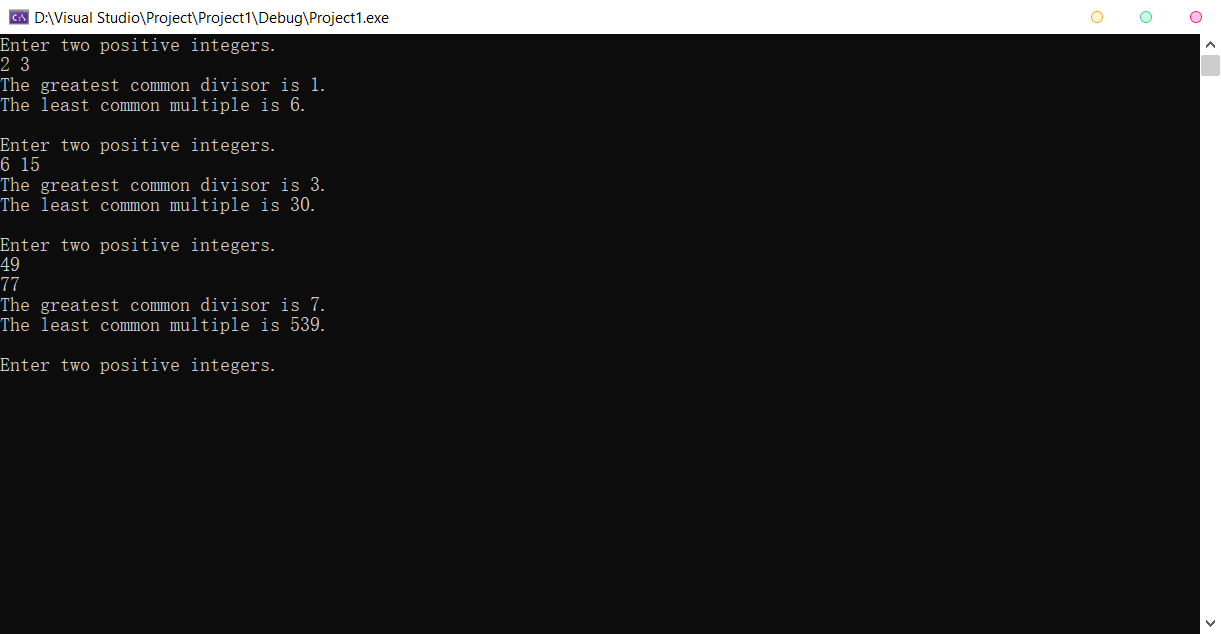
cin >> num1 >> num2;

cout << "The greatest common divisor is " << GCD(num1, num2) << ".\n";

cout << "The least common multiple is " << LCM(num1, num2) << ".\n\n";

system("pause");

return 0;

}

4.

#include<iostream>

#include<string>

using namespace std;

template<typename T>int TwoSmallest(T a, T b, T c, T& s1, T& s2) {

T t;

if (b > c) { t = b; b = c; c = t; }

if (a > b) { t = a; a = b; b = t; }

if (b > c) { t = b; b = c; c = t; }

s1 = a;

if (a == b) {

if (b == c) {

return 3;

}

else {

s2 = c;

return 2;

}

}

else {

s2 = b;

return 1;

}

}

template<typename T>void ShowTwoSmallest(int smallnum, T& s1, T& s2) {

cout << "The smallest number : " << s1 << endl;

if (smallnum != 3)

cout << "The second smallest number : " << s2 << endl;

}

int main() {

begin:

int choice = 0;

cout << "\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n"

<< " 1. int\n 2. double\n 3. char\n 4. string\n 0. exit\n"

<< "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n"

<< "Inout choice: ";

cin >> choice;

int i1 = 0, i2 = 0, i3 = 0, is1 = 0, is2 = 0;

double d1 = 0, d2 = 0, d3 = 0, ds1 = 0, ds2 = 0;

char c1 = 0, c2 = 0, c3 = 0, cs1 = 0, cs2 = 0;

string st1, st2, st3, sts1, sts2;

switch (choice) {

case 1:

cout << "Number1: ";

cin >> i1;

cout << "Number2: ";

cin >> i2;

cout << "Number3: ";

cin >> i3;

ShowTwoSmallest(TwoSmallest(i1, i2, i3, is1, is2), is1, is2);

goto begin;

case 2:

cout << "Number1: ";

cin >> d1;

cout << "Number2: ";

cin >> d2;

cout << "Number3: ";

cin >> d3;

ShowTwoSmallest(TwoSmallest(d1, d2, d3, ds1, ds2), ds1, ds2);

goto begin;

case 3:

cout << "Character1: ";

cin >> c1;

cout << "Character2: ";

cin >> c2;

cout << "Character3: ";

cin >> c3;

ShowTwoSmallest(TwoSmallest(c1, c2, c3, cs1, cs2), cs1, cs2);

goto begin;

case 4:

cout << "String1: ";

cin >> st1;

cout << "String2: ";

cin >> st2;

cout << "String3: ";

cin >> st3;

ShowTwoSmallest(TwoSmallest(st1, st2, st3, sts1, sts2), sts1, sts2);

goto begin;

default:

cout << "Exit\n";

break;

}

system("pause");

return 0;

}

