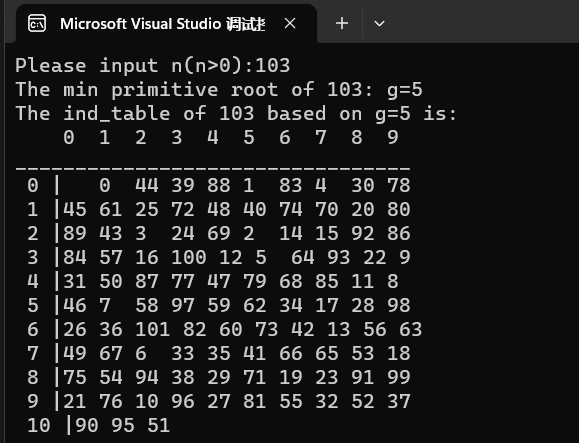
姓名：艾明旭 学号：2111033

本次编程作业的实现思路：提前先求出欧拉函数，之后再通过遍历，我们寻找到其中的满足欧拉函数等于次数的最小值，即为改输入值的最小原根，求出最小原根之后，我们就可以通过求出其缩系。我们对最小原根的缩系求指数。并对其取模，之后就能得到我们所求的指数表。最后再调整一下输出格式，就能得到输出。



#include <iostream>

#include<math.h>

using namespace std;

const int maxn = 100;

long long prime[maxn];

int cnt = 0;

void make\_prime(long long x) {

for (int i = 2; i \* i <= x; i++)

{

if (x % i == 0)

{

prime[cnt++] = i;

while (x % i == 0)

{

x /= i;

}

}

}

if (x > 1)

{

prime[cnt++] = x;

}

}

long long get\_pow(long long x, int n, int mod)

{

long long ans = 1;

while (n)

{

if (n & 1) //n的末尾数字为1

{

ans = ans \* x % mod;

}

x = x \* x % mod;

n >>= 1;//按位右移一位

}

if (ans < 0)

{

ans += mod;

}

return ans;

}

int maxi(int a, int b) {

int i = 0;

while (b != 0) {

i = a % b;

a = b;

b = i;

}

return a;

}//求最大公因数的函数

int main(void) {

cout << "Please input n(n>0):";

long long m;

cin >> m;

//cout << m << endl;

make\_prime(m - 1);

int temp = 0;

for (int i = 2; i < m; i++)

{

int flag = 1;

for (int j = cnt-1;j>=0;j--)

{

int x = (m - 1) / prime[j];

if (get\_pow(i, x, m) == 1) {

flag = 0;

break;

}

}

if (flag)

{

temp = i;

break;

}

}

int g = temp;

cout << "The min primitive root of " << m << ": g=" << temp<<endl;

cout << "The ind\_table of " << m << " based on g=" << temp << " is:"<<endl;

int oula = 0;

for (int t = 1; t < m; t++) {

if (maxi(m, t) == 1) {

oula++;

}

}//求欧拉函数

int r = 0;

int shii = oula / 10;//记录十位数字

shii++;

int\*\* ans = new int\* [shii];

for (int i = 0; i < shii; i++)

ans[i] = new int[10];//声明最后的数组

for (int i = 0; i < shii; i++) {

for (int j = 0; j < 10; j++) {

ans[i][j] = -1;//先把初值全部赋值为-1

}

}

int\* t = new int[oula];//记录g的r次方的同余值

for (r = 0; r < oula; r++) {

if (r == 0) {

t[0] = 1;//欧拉函数为1

}

else {

t[r] = t[r - 1] \* g;

t[r] = fmod(t[r], m);

}

int shi = t[r] / 10;

int ge = t[r] % 10;

ans[shi][ge] = r;

}

cout << " ";//一些输出格式控制语句

for (int i = 0; i < 10; i++) {

cout << i << " ";

}

cout << endl;

cout << "\_\_\_";

for (int i = 0; i < 10; i++) {

cout << "\_\_\_";

}

cout << endl;

for (int i = 0; i < shii; i++) {

for (int j = 0; j < 10; j++) {

if (j == 0) {

cout << " " << i << " |";

}

if (ans[i][j] == -1) {

cout << " ";

}

else {

cout << ans[i][j] << " ";

if (ans[i][j] / 10 == 0) {

cout << " ";

}

}

}

cout << endl;

}

return 0;

}