《C++程序设计基础》课本习题

P122T2:

/\* P122T2 - 使用程序计算数学表达式的值，注意判断溢出的情况 \*/

#include <iostream>

#include <cmath>

using namespace std;

int main()

{

long long m, n, p;

while (cout << "n, m, p: ", cin >> n >> m >> p)

{

// 利用组合数学公式简化运算

long long s1 = (1 + n) \* n / 2;

long long s2 = m \* m \* (m + 1) \* (m + 1) / 4;

long long s3 = p \* p \* (p + 1) \* (p + 1) \* (2 \* p \* p + 2 \* p - 1) / 12;

// 判断溢出

if (s1 < 0 || s2 < 0 || s3 < 0) // 不能检测出所有溢出情况

{

cout << "Overflow!" << endl;

continue;

}

cout << "Result: " << (double)(s1 + s2) / s3 << endl;

}

}

P122T6:

/\* P122T6 - 将利用循环的程序改写成等效的递归程序 \*/

#include <iostream>

using namespace std;

void print(int w, bool endline = true)

{

if (w == 1) {

if (endline) cout << 1 << endl;

else cout << 1 << " ";

}

else {

if (endline) {

print(w - 1);

print(w - 1, false);

cout << w << endl;

}

else {

print(w - 1, false);

cout << w << " ";

}

}

}

int main()

{

print(5);

}

P122T8:

/\* P122T8 - 利用重载函数编程序分别把两个数和三个数从大到小排列 \*/

#include <iostream>

using namespace std;

void swap(int &a, int &b)

{

a = a ^ b;

b = a ^ b;

a = a ^ b;

}

void sort(int a, int b)

{

if (a > b) cout << a << " " << b << endl;

else cout << b << " " << a << endl;

}

void sort(int a, int b, int c)

{

if (a < b) swap(a, b);

if (a < c) swap(a, c);

if (b < c) swap(b, c);

cout << a << " " << b << " " << c << endl;

}

int main()

{

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

{

if (i % 2) sort(rand(), rand(), rand());

else sort(rand(), rand());

}

}

P168T3:

/\* P168T3 - 将数组元素按非升序排列，输出排序后的每个元素及其在原数组中的下标 \*/

#include <iostream>

#include <cassert>

#include <ctime>

using namespace std;

void swap(int&, int&);

void sort(int[], int);

int main()

{

srand(time(0));

for (int i = 1; i <= 3; i++)

{

if (i != 1) cout << endl;

cout << "Case #" << i << ": " << endl;

int length = rand() % 10 + 1;

int \* num = new int[length]; assert(num != NULL);

for (int i = 0; i < length; i++) num[i] = rand();

sort(num, length);

}

}

void swap(int &a, int &b)

{

a = a + b;

b = a - b;

a = a - b;

}

void sort(int num[], int length)

{

int \*idx = new int[length]; assert(idx != NULL);

for (int i = 0; i < length; i++) idx[i] = i;

// 输出原数组

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

{

if (i) cout << " ";

cout << num[i];

}

cout << endl;

// 排序

for (int i = 0; i != length - 1; i++)

{

for (int j = length - 1; j > i; j--)

{

if (num[j] > num[j - 1])

swap(num[j], num[j - 1]), swap(idx[j], idx[j - 1]);

}

}

// 输出排序后的数组元素及其在原数组中的下标

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

{

if (i) cout << " ";

cout << num[i] << "(" << idx[i] << ")";

}

cout << endl;

delete idx;

}

P168T4:

/\* P168T4 - 回文数判定 \*/

#include <iostream>

using namespace std;

int main()

{

long long num;

while (cin >> num)

{

long long replica = num, mirror = 0;

while (replica)

{

mirror = mirror \* 10 + replica % 10;

replica /= 10;

}

if (mirror == num) cout << "Yes" << endl;

else cout << "No" << endl;

}

}

P168T13:

#include <iostream>

#include <string>

using namespace std;

int serial[] = {1, 2, 3, 4, 5, 6};

string name[] = {"Kehao", "Shanxin", "Lightyears", "Guoyun", "Bill", "Zekar"};

int main()

{

int id;

cout << "Please give me a serial numeber: " << endl;

cin >> id;

{

int i;

for (i = 0; i < 6; i++) if(id == serial[i])

{

cout << name[i] << endl;

break;

}

if (i == 6) cout << "I can't find a name with the given serial." << endl;

}

}