ĐÁP ÁN BUỔI 11 – HÀM VÀ THAM SỐ

1. Viết hàm cho phép truyền vào tham số n và trả về tổng sau:

Trong hàm main, cho phép nhập n từ bàn phím và xuất ra kết quả tổng S bằng cách gọi hàm đã khai báo ở trên.

#include<iostream>

using namespace std;

int TinhTong(int n)

{

int S = 0;

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

S = S + i \* i;

return S;

}

int main()

{

int n;

cout << "Nhap n: ";

cin >> n;

int tong = TinhTong(n);

cout << "Tong = " << tong;

}

1. Viết hàm cho phép truyền vào tham số n và trả về tổng sau:

Trong hàm main, cho phép nhập n từ bàn phím và xuất ra kết quả tổng S bằng cách gọi hàm đã khai báo ở trên.

#include<iostream>

using namespace std;

float TinhTong(int n)

{

float S = 0;

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

S = S + 1.0 / i; // hoac 1 / (float)i

return S;

}

int main()

{

int n;

cout << "Nhap n: ";

cin >> n;

float tong = TinhTong(n);

cout << "Tong = " << tong;

}

1. Viết hàm cho phép truyền vào tham số n và trả về tổng sau:

Trong hàm main, cho phép nhập n từ bàn phím và xuất ra kết quả tổng S bằng cách gọi hàm đã khai báo ở trên.

#include<iostream>

using namespace std;

float TinhTong(int n)

{

float S = 0;

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

S = S + (float)i / (i + 1); // hoac i / (float)(i + 1)

return S;

}

int main()

{

int n;

cout << "Nhap n: ";

cin >> n;

float tong = TinhTong(n);

cout << "Tong = " << tong;

}

1. Viết hàm cho phép truyền vào số nguyên dương, trả về tổng các chữ số của số nguyên này. Trong hàm main, cho phép nhập số nguyên dương n từ bàn phím, nếu n không phải số nguyên dương thì yêu cầu nhập lại đến khi nào thỏa mãn điều kiện, gọi hàm đã khai báo ở bước trên để tính tổng và xuất kết quả.

Ví dụ: n = 25147, tổng các chữ số = 2 + 5 + 1 + 4 + 7 = 19.

#include<iostream>

using namespace std;

int TongChuSo(int n)

{

int S = 0;

while (n > 0)

{

int chuSoCuoi = n % 10;

S += chuSoCuoi;

n /= 10; // n = n / 10

}

return S;

}

int main()

{

int n;

do

{

cout << "Nhap n: ";

cin >> n;

} while (n <= 0);

int tong = TongChuSo(n);

cout << "Tong = " << tong;

}

1. Viết hàm cho phép truyền vào 3 số thực, trả về số lớn nhất trong 3 số này. Trong hàm main cho phép nhập 3 số thực a, b, c từ bàn phím. Xuất ra số lớn nhất bằng cách gọi hàm đã khai báo ở trên.

#include<iostream>

using namespace std;

float FindMax(float a, float b, float c)

{

float max = a;

if (b > max)

max = b;

if (c > max)

max = c;

return max;

}

int main()

{

float a, b, c;

cout << "Nhap a, b, c: ";

cin >> a >> b >> c;

float result = FindMax(a, b, c);

cout << "So lon nhat la: " << result;

}

1. Viết hàm cho phép truyền vào 2 số nguyên tương ứng với tháng và năm. Trả về số ngày của tháng đó (tham khảo cách làm từ bài học trước). Trong hàm main cho phép nhập tháng, năm, xuất ra số ngày của tháng đó bằng cách gọi hàm đã khai báo ở trên.

Cách 1:

#include<iostream>

using namespace std;

int SoNgayTrongThang(int month, int year)

{

int soNgay;

switch (month)

{

case 1:case 3: case 5: case 7: case 8: case 10: case 12:

soNgay = 31;

break;

case 2:

// kiem tra nam nhuan

if ((year % 4 == 0 && year % 100 != 0) || year % 400 == 0)

soNgay = 29;

else

soNgay = 28;

break;

default: // cac thang con lai

soNgay = 30;

break;

}

return soNgay;

}

int main()

{

int m; // month

int y; // year;

cout << "Nhap thang, nam: ";

cin >> m >> y;

if (m < 1 || m > 12)

cout << "Thang khong hop le";

else

{

int soNgay = SoNgayTrongThang(m, y);

cout << "So ngay trong thang " << m << " nam "

<< y << " = " << soNgay;

}

}

Cách 2: sử dụng lệnh return

#include<iostream>

using namespace std;

int SoNgayTrongThang(int month, int year)

{

switch (month)

{

case 1:case 3: case 5: case 7: case 8: case 10: case 12:

return 31;

case 2:

// kiem tra nam nhuan

if ((year % 4 == 0 && year % 100 != 0) || year % 400 == 0)

return 29;

return 28;

default: // cac thang con lai

return 30;

}

}

int main()

{

int m; // month

int y; // year;

cout << "Nhap thang, nam: ";

cin >> m >> y;

if (m < 1 || m > 12)

cout << "Thang khong hop le";

else

{

int soNgay = SoNgayTrongThang(m, y);

cout << "So ngay trong thang " << m << " nam "

<< y << " = " << soNgay;

}

}

Chú ý: lệnh return sẽ trả về kết quả và thoát khỏi hàm ngay sau đó.

1. Viết chương trình thực hiện các yêu cầu sau:
2. Viết hàm cho phép truyền vào số nguyên dương n, kiểm tra xem n có phải là số nguyên tố (tham khảo bài học trước, **hàm trả về kiểu bool,** nếu nguyên tố thì trả về **true**, ngược lại thì trả về **false**).
3. Viết hàm cho phép truyền vào số nguyên dương n, in ra các số nguyên tố trong đoạn từ 1 đến n.

Gợi ý: sử dụng hàm đã khai báo ở câu a.

1. Trong hàm main, cho phép người dùng nhập số nguyên dương n từ bàn phím, gọi hàm đã khai báo ở câu b để xuất ra danh sách số nguyên tố trong đoạn từ 1 đến n.

Ví dụ: nhập n = 20, xuất ra danh sách các số nguyên tố trong đoạn [1, 20] như sau:

2 3 5 7 11 13 17 19

#include<iostream>

#include<cmath>

using namespace std;

// cau a

// ham kiem tra xem n co phai so nguyen to khong

bool NguyenTo(int n)

{

if (n <= 1)

return false;

// truong hop n > 1 thi xu li tiep nhu sau

for (int i = 2; i <= sqrt(n); i++)

{

if (n % i == 0)

return false;

}

// n khong chia het so so nguyen nao trong doan [2, sqrt(n)]

return true;

}

// cau b

// xuat danh sach so nguyen to trong doan tu 1 den n

void XuatDaySoNguyenTo(int n)

{

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

{

if (NguyenTo(i) == true)

cout << i << " ";

}

}

int main()

{

int n;

cout << "Nhap n: ";

cin >> n;

cout << "Danh sach cac so nguyen to tu 1 den " << n << ":\n";

XuatDaySoNguyenTo(n);

}

Ghi chú: có thể giải theo nhiều cách khác nhau, như đã sửa ở bài trước.