**BÀI TẬP LỆNH CẤU TRÚC ĐIỀU KHIỂN**

**Kết thúc bài thực hành này bạn có khả năng**

* Sử dụng vòng lặp cho biết trước số lần lặp for
* Sử dụng vòng lặp chưa biết trước số lần lặp: While, do…while
* Sử dụng một số lặp: continue, break

**Phần 2: Cấu trúc lặp: while, do..while, for**

**Bài 4:** Viết chương trình nhập vào 2 số min và max từ bàn phím. Tính và xuất trung bình cộng các số chia hết cho 5 từ min đến max.

**HƯỚNG DẪN:**

* Nhập 2 số nguyên từ bàn phím
* Thực hiện vòng lặp while từ min đến max int i = min;

double tong = 0, dem = 0, trungBinh = 0;

while(i <= max){…}

* Trong while viết mã cộng i vào tổng nếu i chia hết cho 5 if(i % 5 == 0){

tong += i;

dem=dem+1;

}

* Tính trung bình cộng trungBinh = tong/dem

**Bài 5:** Viết chương trình nhập một số nguyên từ bàn phím và cho biết số đó có phải là số nguyên tố hay không (số nguyên tố là số chỉ chia hết cho 1 và chính nó).

**HƯỚNG DẪN**

* Nhập số nguyên N từ bàn phím
* Cho một vòng lặp chạy từ 2 cho đến số nhập vào/2. boolean soNguyenTo = true;

for(int i=2; i < N/2; i++){ …}

* Trong vòng lặp for nếu số nhập vào chia hết cho i thì số đó không phải là số nguyên tố.

if(N % i == 0){

soNguyenTo = false; break;

}

Sau vòng lặp for kiểm tra biến soNguyenTo bạn sẽ biết N có phải là số nguyên tố hay không

**Bài 6:** Viết chương trình nhập từ bàn phím số m. Xuất dãy số fibonaci các số nhỏ hơn m. Dãy fibonaci là dãy số có số sau bằng tổng 2 số đứng kế trước trong đó 2 số đầu

tiên f0=1 và f1=2.

**HƯỚNG DẪN**

* Nhập số m từ bàn phím
* Chuẩn bị các biến cần thiết và

int f0 = 1, f1 = 2, next = f0 + f1;

* Xuất dãy fibonaci

while(next < m){

next = f0 + f1;

Xuất next ra màn hình f0=f1

f1=next

}

**Bài 7**: Viết chương trình liệt kê các số nguyên dương có 3 chữ số và có từ 20 ước số trở lên.

**HƯỚNG DẪN**

Cho vòng for chạy từ 100 đến 999. Với mỗi số i trong phạm vi này, ta chia i cho các số j (với j chạy từ 1 đến i) và đếm số ước số của i (j là ước số của i nếu i%j==0).

**Bài toán mở rộng**

Có 1 triệu đồng, hỏi phải gửi tiết kiệm vào ngân hàng trong thời gian mấy năm để có số tiền lớn hơn 2 triệu đồng. Biết rằng lãi suất hàng năm là 8%.

**HƯỚNG DẪN**

- Vì không thể biết được là sau bao nhiêu năm thì tổng số tiền lớn hơn 2.000.000, mà chỉ biết rằng mỗi năm sẽ tăng thêm một lượng nào đó. Do vậy, ở đây ta sẽ sử dụng vòng lặp không xác định và mỗi lần lặp ta sẽ kiểm tra xem đã được số tiền cần thiết hay chưa? Nếu đủ rồi thì thoát và số lần thử chính là số năm cần tìm.

Nhưng ở đây tại sao ta lại sử dụng vòng lặp while mà không là do …while? Sở dĩ sử dụng vòng lặp while là vì số tiền gửi vào ban đầu rất có thể đã lớn hơn số tiền kỳ vọng!

- Tính theo công thức lãi kép, lãi nhập vốn:

1. Lãi = gốc\*(1+lãi suất)^n - gốc

2. Số tiền rút ra = gốc\*(1+lãi suất kỳ)^n. (công thức của FV) với n là số kỳ

- Sử dụng hàm Math.pow (x, y) để tính x^y, hàm Math.round(x) để làm tròn số thực x.

**Bài 8:** Viết chương trình nhập số N sau đó tính các tổng sau S1=1 + 2 + 3 +. + N

S2=1 +1/2+1/3+. +1/N

S3=1 +22+33+. +NN

S4=1\*2\*3. \*N

S5= 1 + 1/2! + 1/3! + + 1/N!

S6= 1/(1\*2) + 1/(2\*3) + 1/(3\*4) + ..... + 1/(N\*(N+1))

**Bài 9:** Viết chương trình tính tổng bình phương các số lẻ từ 1 đến N.

**Bài 10:** Viết chương trình tính số hạng thứ n của dãy Fibonaci.

Dãy Fibonaci là dãy số gồm các số hạng p(n) với: p(n) = p(n-1) + p(n-2) với n>2 và p(1) = p(2) = 1

p(3) = p(3-1) + p(3-2) = p(2) + p(1) = 2

p(4) = p(4-1) + p(2-2) = p(3) + p(2) = p(2) + p(1) + p(2) =3

Dãy Fibonaci sẽ là: 1 1 2 3 5 8 13 21 34 55 89 144…