I. Thuật toán

-Khái niệm: là một bản liệt kê các chỉ dẫn, các quy tắc cần thực hiện theo từng bước xác định nhằm giải quyết một bài toán đã cho trong một khoảng thời gian hữu hạn

-Tính chất của thuật toán:

+ Tính chính xác: để đảm bảo kết quả tính toán hay các thao tác mà máy tính thực hiện được là chính xác

+ Tính rõ ràng: Thuật toán phải được thể hiện bằng các câu lệnh minh bạch; các câu lệnh được sắp xếp theo thứ tự nhất định

+ Tính khách quan: Một thuật toán dù được viết bởi nhiều người trên nhiều máy tính vẫn phải cho kết qủa như nhau

+ Tính phổ dụng: thuật toán không chỉ áp dụng cho 1 bài toán nhất định mà có thể áp dụng cho 1 lớp các bài toán có đầu vào tương tự nhau

+ Tính kết thúc: Thuật toán phải gồm 1 số hữu hạn các bước tính toán

-Biểu diễn thuật toán

+ Dùng ngôn ngữ tự nhiên

+ Sơ đồ khối

+ Giả mã

VD: Tìm phần tử có giá trị B trong mảng

Input: I0 ,mảng a[0],…,a[n]

Output: nếu B có trong mảng => vị trí của B trong mảng ngược lại in ra -1

1, I0 = 0

2, for(i=0;i<=n;i++)

If(a[i]==’B’)

I0=i+1;

Else

I0= -1;

2, Output I0

-Độ phức tạp thuật toán:

+Quy tắc bỏ hằng số

T(n)=O(c.f(n))=O(f(n)) với c là một hằng số dương

+Quy tắc lấy max

T(n)= O(f(n)+g(n))=O(max(f(n),g(n)))

+Quy tắc cộng

T1(n)=O(f(n)) T2(n)=O(g(n))

T1(n)+T2(n)=O(f(n)+g(n))

+Quy tắc nhân

++Đoạn chương trình có thời gian thực hiện T(n)=O(f(n))

++Nếu thực hiện k(n) lần đoạn chương trình với k(n)=)(g(n)) thì độ phức tạp là O(f(n).g(n))

-Thứ tự độ phức tạp của thuật toán:

O(log n),O(1)<O(n)<O(n log n)<O(n2)<O(2^n)<O(n!)

VD1: s=n\*(n-1)/2;

* O(1)

VD2:

Bài tập:

1. Nêu định nghĩa, tính chất và các cách thức biểu diễn thuật toán
2. Cho bài toán sau:

a, Tính nghiệm phương trình bậc 2: ax2+bx+c=0, a#0.

b, Tính tổng bình phương của n số tự nhiên đầu tiên.

c, Tìm số có giá trị x trong dãy x1,x2,…,xn.

d, Tìm số có giá trị lớn nhất trong dãy x1,x2,…,xn.

Hãy tìm thuật toán để giải bài toán trên, mô tả các thuật toán sử dụng ngôn ngữ tự nhiên và chỉ ra các tính chất của thuật toán đó

b, Input: n

Output: S(n) tổng bình phương của n số tự nhiên đầu tiên

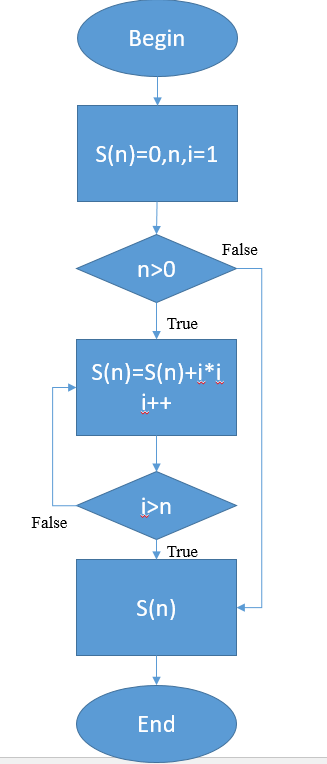
1, S(n)=0

2, for(i=1;i<=n;i++)

S(n)+=i\*i

3, S(n)

-Sơ đồ khối:



2.Đệ quy và giải thuật đệ quy

-Khái niệm: một đối tượng là đệ quy nếu nó được định nghĩa qua chính nó hoặc một đối tượng khác cùng dạng với nó

-Giải thuật đệ quy:

+Một thủ tục đệ quy gồm 2 phần:

++Phần neo: là phần đơn giản có thể thực hiện trực tiếp

++Phần đệ quy

Bài tập đệ quy:

BT1: Tính X lũy thừa n

BT2: Thuật toán Euclide tìm ƯCLN

BT3: Tính n giai thừa

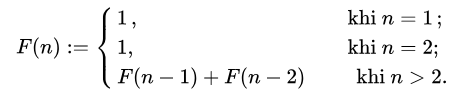
BT4: Viết phương trình tính tổng n phần tử a1,a2,…,an

BT5: Tích n phần tử trong danh sách

BT6: Tìm giá trị lớn nhất, nhỏ nhất của mảng gồm n phần tử

-Fibonacci:

**Dãy Fibonacci** là [dãy](https://vi.wikipedia.org/wiki/D%C3%A3y_(to%C3%A1n_h%E1%BB%8Dc)) [vô hạn](https://vi.wikipedia.org/wiki/V%C3%B4_t%E1%BA%ADn) các [số tự nhiên](https://vi.wikipedia.org/wiki/S%E1%BB%91_t%E1%BB%B1_nhi%C3%AAn) bắt đầu bằng hai phần tử 0 và 1 hoặc 1 và 1, các phần tử sau đó được thiết lập theo quy tắc *mỗi phần tử luôn bằng tổng hai phần tử trước nó*. [Công thức truy hồi](https://vi.wikipedia.org/w/index.php?title=C%C3%B4ng_th%E1%BB%A9c_truy_h%E1%BB%93i&action=edit&redlink=1) của dãy Fibonacci là:

{\displaystyle F(n):=\left\{{\begin{matrix}1\,,\qquad \qquad \qquad \quad \,\ \ \,&&{\mbox{khi }}n=1\,;\ \ \\1,\qquad \qquad \qquad \qquad \,&&{\mbox{khi }}n=2;\ \ \,\\F(n-1)+F(n-2)&&{\mbox{khi }}n>2.\end{matrix}}\right.}

-Tòa tháp hà nội: 2n-1 bước

