

# THUẬT TOÁN ỨNG DỤNG

## CẤU TRÚC DỮ LIỆU VÀ THƯ VIỆN

Phạm Quang Dũng  
Bộ môn KHMT  
[dungpq@soict.hust.edu.vn](mailto:dungpq@soict.hust.edu.vn)

# Nội dung

---

- Danh sách tuyến tính
- Tập hợp
- Ánh xạ
- Ngăn xếp
- Hàng đợi
- Sắp xếp

# Danh sách tuyến tính

---

- Lưu trữ các đối tượng theo quan hệ tuyến tính (trước – sau)
- Thao tác: thêm, xóa, tìm kiếm

# List

```
#include <bits/stdc++.h>
using namespace std;
int main(){
    list<int> L;
    for(int i = 1; i<=5;i++){
        L.push_back(i);
    }
    list<int>::iterator it;
    it = find(L.begin(),L.end(),3);
    L.insert(it,10);
    for(it = L.begin(); it != L.end(); it++){
        cout << *it << endl;
    }
}
```

# Vector

```
#include <bits/stdc++.h>
using namespace std;
int main(){
    vector<int> V(3,100); // initialize 3 elements 100
    for(int v = 0; v <= 10; v++)
        V.push_back(v);

    cout << "vector: ";
    for(int i = 0; i < V.size(); i++){
        cout << V[i] << " ";
    }
}
```

# Tập hợp

---

- Lưu các đối tượng, không trùng nhau
- Thao tác: thêm, xóa, tìm kiếm

# Tập hợp

```
#include <bits/stdc++.h>
using namespace std;
int main(){
    set<int> Y;
    for(int i = 1; i <= 10; i++){
        Y.insert(i);
    }
    for(set<int>::iterator it = Y.begin(); it != Y.end(); it++){
        cout << *it << endl;
    }
    if(Y.find(7) != Y.end())
        cout << "Y contains 7" << endl;
    else
        cout << "Y does not contains 7" << endl;
}
```

# Ảnh xạ

---

- Cấu trúc dữ liệu cất trữ các cặp (khóa, giá trị)
- Phục vụ tìm kiếm nhanh với khóa đầu vào



# Ảnh xạ

```
#include <bits/stdc++.h>
using namespace std;
int main(){
    map<int,int> m;
    for(int i = 1; i <= 5; i++)
        m.insert(pair<int,int>(i,10*i));
    m[6] = 100;
    for(int k = 1; k <= 6; k++) cout << m[k] << endl;

    map<string, string> m1;
    m1["abc"] = "abcabc";
    m1["xyz"] = "xyzxyz";
    string s = "abc";
    cout << m1[s] << endl;
}
```

# Ảnh xạ

```
#include <bits/stdc++.h>
using namespace std;
int main(){
    map<pair<int,int>, pair<int,int> > m2;
    m2[pair<int,int>(2,5)] = pair<int,int>(20,50);
    m2[pair<int,int>(3,5)] = pair<int,int>(30,50);

    int i = 3;
    int j = 5;
    pair<int,int> p = m2[pair<int,int>(i,j)];
    cout << p.first << "," << p.second << endl;
}
```

# Ngăn xếp

---

- Cấu trúc dữ liệu cất trữ các đối tượng một cách tuyến tính
- Thao tác
  - Thêm 1 phần tử
  - Lấy ra 1 phần tử
- Nguyên tắc: Vào trước – ra sau

# Hàng đợi

---

- Cấu trúc dữ liệu cất trữ các đối tượng một cách tuyến tính
- Thao tác
  - Thêm 1 phần tử
  - Lấy ra 1 phần tử
- Nguyên tắc: vào trước – ra trước

# Stack

---

```
#include <bits/stdc++.h>
using namespace std;
int main(){
    stack<int> S;
    for(int i = 0; i < 5; i++){
        S.push(i);
    }
    while(!S.empty()){
        int v = S.top(); S.pop();
        cout << v << endl;
    }
}
```

# Queue

---

```
#include <bits/stdc++.h>
using namespace std;
int main(){
    queue<int> Q;
    for(int i = 0; i < 5; i++){
        Q.push(i);
    }
    while(!Q.empty()){
        int v = Q.front(); Q.pop();
        cout << v << endl;
    }
}
```

# Sắp xếp

```
#include <algorithm>
#include <iostream>
using namespace std;
int main(){
    int N = 6;
    double a[N] = {1.1, 5.5, 7.7, 2.2, 8.8, 3.3};
    sort(a+3,a+N,greater<double>()); // decreasing order
    for(int i = 0; i < N; i++) cout << a[i] << " ";
    cout << endl;
    sort(a,a+N);
    for(int i = 0; i < N; i++) cout << a[i] << " ";
}
```