**Bài 1: Marathon**

- Dùng cấu trúc dữ liệu mảng để lưu thông tin nhập vào

- Bóc tách dữ liệu nhập thành 3 phần, dùng hàm rsplit()

- Lưu riêng thông tin của các sinh viên khoa DDT và các khoa còn lại vào 2 mảng

- Sắp xếp 2 mảng đó theo chiều giảm dần về điểm

- In ra kết quả

**Bài 2: Liệt kê những số đi được trong bài mọi con đường về 0**

* Cấu trúc dữ liệu: queue
* Ý tưởng:

+ Phân tách số n thành các cặp tích a \* b, từ cặp (a,b) sinh ra số tự nhiên m = (a-1)\*(b+1) và cho vào hàng đợi

+ Lấy phần tử đầu tiên trong hàng đợi ra và tiếp tục phân tách số vừa lấy cho đến khi các số con = 0 thì dừng lại.

**Bài 3: Đong nước:**

* Cấu trúc dữ liệu: queue
* Ý tưởng:

+ Cho 2 bình nước chứa tối đa n và m lít, ta có các trường hợp có thể đong nước:

TH1: Đổ đầy bình nước 1

TH2: Đổ đầy bình nước 2

TH3: Đổ bỏ nước ở bình 1

TH4: Đổ bỏ nước ở bình 2

TH5: Đổ nước từ bình 1 sang bình 2

TH6: Đổ nước từ bình 2 sang bình 1

+ Ta xét các trường hợp đổ nước thỏa mãn, mà khối lượng nước của 2 bình không bị trùng lặp với khối lượng nước của 2 bình đã xét trước đó. Nếu 1 trong 2 bình có khối lượng nước trùng với khối lượng nước cần đong thì trả về số bước đã đong. Còn không trùng ta vẫn tiếp tục đong nước và cộng dồn số bước đi. Nếu hết trường hợp để xét, ta kết luận “Khong dong duoc nuoc”

**Bài 4: Phân vùng ảnh**

* Cấu trúc dữ liệu: queue
* Ý tưởng:

+ Duyệt ma trận nếu phần tử đang xét có giá trị là 0 thì duyệt theo thuật toán BFS

* Đánh dấu phần tử đang xét là đã được thăm và cho vào hàng đợi.
* Duyệt 8 phần tử kề với phần tử đang xét nếu thỏa mãn thì đánh dấu là đã thăm và cho vào hàng đợi đồng thời cập nhật số lượng tăng lên 1.
* Tiếp tục duyệt cho đến khi hàng đợi rỗng.

**Bài 5: Bài toán giao hàng**

* Cấu trúc dữ liệu: priority queue (hàng đợi ưu tiên)
* Ý tưởng:

+ Nhóm các món hàng có cùng thời hạn giao hàng giống nhau

+ Duyệt từ mốc hạn giao hàng cuối cùng về 1. Thêm tất cả các món hàng có hạn trong khoảng thời gian đang xét vào trong hàng đợi ưu tiên (sắp xếp theo giá trị phần thưởng). Nếu hàng đợi không rỗng thì lấy món hàng có phần thưởng lớn nhất và loại bỏ khỏi hàng đợi.

**Bài 6: Trinh thám**

* Cấu trúc dữ liệu: priority queue (hàng đợi ưu tiên)
* Ý tưởng:

+ Duyệt lần lượt từng phần tử:

* Cho 1 cặp giá trị chiều cao và vị trí vào hàng đợi ưu tiên (sắp xếp theo giá trị chiều cao giảm dần).
* Nếu vị trí phần tử đang xét >= k - 1thì: kiểm tra trong hàng đợi ưu tiên nếu vị trí của phần tử đầu tiên không thuộc khoảng đang xét thì bỏ khỏi hàng đợi, tiếp tục lặp lại cho đến khi phần tử có vị trí thỏa mãn thì in ra màn hình.

**Bài 7: Truy vấn max của đoạn con liên tiếp**

* Cấu trúc dữ liệu: Cây IT (interval tree)
* Ý tưởng:

+ Xây dựng một cây Interval Tree để quản lý tập các khoảng [l, r], mỗi nút của cây lưu giá trị lớn nhất tại khoảng l, r.

+ Nếu nút cha lưu giá trị lớn nhất của đoạn [l, r] thì:

* Nút con trái sẽ lưu giá trị lớn nhất của đoạn [l, (l+r)/2]
* Nút con phai sẽ lưu giá trị lớn nhất của đoạn [(l+r)/2 + 1, r]

+ Có 2 truy vấn:

* Truy vấn 1: Cập nhật giá trị của các nút trên cây mà đoạn nó quản lý chứa phần tử i (i là phần tử đang xét).
* Truy vấn 2: Tìm các nút trên cây mà đoạn nó quản lý thuộc đoạn [i, j] (với i, j là đoạn đang xét), sau đó lấy max của các nút này

**Bài 8: Đếm số nghịch thế của phép thế**

* Cấu trúc dữ liệu: Cây IT (Interval Tree)
* Ý tưởng:

+ Xây dựng một cây Interval Tree để quản lý tập các khoảng [l, r], mỗi nút của cây lưu giá trị tần xuất của các phần tử trong mảng.

+ Xét lần lượt các phần tử có trong mảng, với mỗi phần tử ta làm:

* Nếu phần tử thuộc khoảng [l, r] mà nút đang quản lý, thì ta tăng giá trị nút thêm 1
* Nếu phần tử thuộc khoảng [l, (l+r)/2] mà nút con trái đang quản lý thì số cặp kết quả cộng thêm giá trị của nút con bên phải (tần xuất những phần tử lớn hơn mà xuất hiện trước đó).

**Bài 9: Nhóm bạn**

* Cấu trúc dữ liệu sử dụng: Disjoinset
* Ý tưởng:

+ Ban đầu có n sinh viên tương ứng với n khu vực, m mối quan hệ

+ Xét cặp quan hệ của từng sinh viên:

* Nếu hai sinh viên không cùng khu vực
  + Gộp khu vực có sinh viên ít hơn sang khu vực có sinh viên nhiều hơn
  + Tổng sinh viên trong khu vực mới sẽ bằng tổng sinh viên ở 2 khu vực
  + Giảm số lượng khu vực đi 1

+ Cập nhật lại max (khu vực có số lượng lớn nhất)

**Bài 10: Cây khung nhỏ nhất**

* Cấu trúc dữ liệu sử dụng: Disjoinset
* Ý tưởng:

+ Sắp xếp các cạnh theo trọng số tăng dần

+ Với mỗi đỉnh, ta lưu lại đỉnh cha của nó

+ Xét các cạnh trong đồ thị:

* Nếu 2 đỉnh không cùng cha thì:
  + Gộp cây có số lượng đỉnh ít hơn sang cây có số lượng đỉnh nhiều hơn.
  + Cập nhật 2 đỉnh về chung một đỉnh cha.
  + Lưu lại tổng các trọng số

**Bài 11: Chào đón sinh viên k59**

* Cấu trúc dữ liệu sử dụng: Stack
* Ý tưởng:

+ Khi sinh viên thứ i xếp hàng, thì thực hiện các thao tác sau:

1. Xếp vào cuối hàng

2. Nếu sinh viên đứng đầu hàng thì ghi lại số -1. Mặt khác, nếu sinh viên đứng trước thấp hơn hoặc bằng, thì đuổi sinh viên trước đó ra khỏi hàng. Lặp lại quá trình đuổi cho tới khi sinh viên này đứng đầu hàng, hoặc có sinh viên trước đó cao hơn. Ghi lại vị trí sinh viên đứng trước, hoặc -1 nếu đứng đầu hàng.

**Bài 12: Hậu tố Balan**

* Cấu trúc dữ liệu: stack
* Ý tưởng:

+ Tạo một sâu hậu tố rỗng

+ Xét từng phần tử ký tự trong chuỗi. Nếu ký tự đang xét:

* Là số: thì thêm vào sâu hậu tố
* Là “(“: thì thêm vào stack
* Là “)”: xét từng phần tử trong stack, đưa phần tử sang xâu hậu tố cho đến khi gặp phần tử “(” thì dừng và xóa ký tự “(“ ra khỏi stack

Là phép toán: thì duyệt các phần tử trong stack, nếu phần tử trong stack có độ ưu tiên lớn hơn hoặc bằng ký tự đang xét thì đồng thời xóa và thêm ký tự trong stack đó vào trong sâu hậu tố, cuối cùng thêm ký tự đang xét vào trong stack.

**Bài 13: Xếp hàng**

* Cấu trúc dữ liệu sử dụng: stack
* Ý tưởng:

+ Khi sĩ tử thứ i xếp hàng, thì thực hiện các thao tác sau:

1. Xếp vào cuối hàng

2. Nếu sĩ tử đứng đầu hàng thì không nhìn thấy ai thì ghi lại số 0. Mặt khác, nếu sĩ tử trước thấp hơn, thì và đuổi sĩ tử trước đó ra khỏi hàng.

**Bài 14: Hoán vị lặp**

* Cấu trúc dữ liệu sử dụng: đệ quy, quay lui
* Ý tưởng:

**+** Sắp sếp lại thứ tự các chữ cái tăng dần theo bảng chữ cái

**+** Mỗi chữ cái được xây dựng bằng cách thử tất cả các chữ cái có hợp lệ.

**+** Các bước trong việc liệt kê các hoán vị hợp lệ theo dạng a1a2a3…an:

* Xét tất cả các ký tự a1 có thể nhận, thử a1 nhận tất các ký tự đó. Với mỗi giá trị của a1 ta sẽ:
* Xét tất cả các ký tự của a2 có thể nhận, thử a2 nhận tất cả các ký tự đó. Với mỗi giá trị của a2 ta sẽ…Tiếp tục lặp lại việc xét cho đến khi đủ số ký tự thì in ra màn hình.

**Bài 15: Lai ghép**

* Cấu truc dữ liệu sử dụng: đệ quy, quay lui
* Ý tưởng:

**+** Với mỗi vị trí, so sánh 2 sâu ký tự:

* Nếu 2 ký tự giống nhau: lấy luôn ký tự đó cho vào sâu kết quả và duyệt tiếp các ký tự đằng sau và in ra kết quả.
* Nếu 2 ký tự khác nhau: lấy ký tự bé hơn (theo bảng chữ cái) để cho vào sâu kết quả trước, sau khi duyệt xong và đã in ra các sâu kết quả thỏa mãn, thì ta thay ký tự có giá trị bé hơn bằng ký tự có giá trị lớn hơn.