アルゴ 2022

1 (1) $g(n) = n^3$

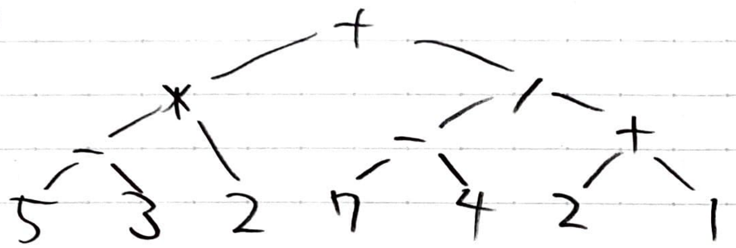
(2) n^2

(3) n^n

$f(n) = O(g(n))$

$\overset{def}{\Longleftrightarrow} \exists C > 0, N_0 > 0, \forall N \geq N_0, \left|\frac{f(N)}{g(N)}\right| \leq C$

2 (1)



(2) 5 3 − 2 * 7 4 − 2 1 + / +

3 再帰が終わった順に出力したい → 帰りがけ順のDFS

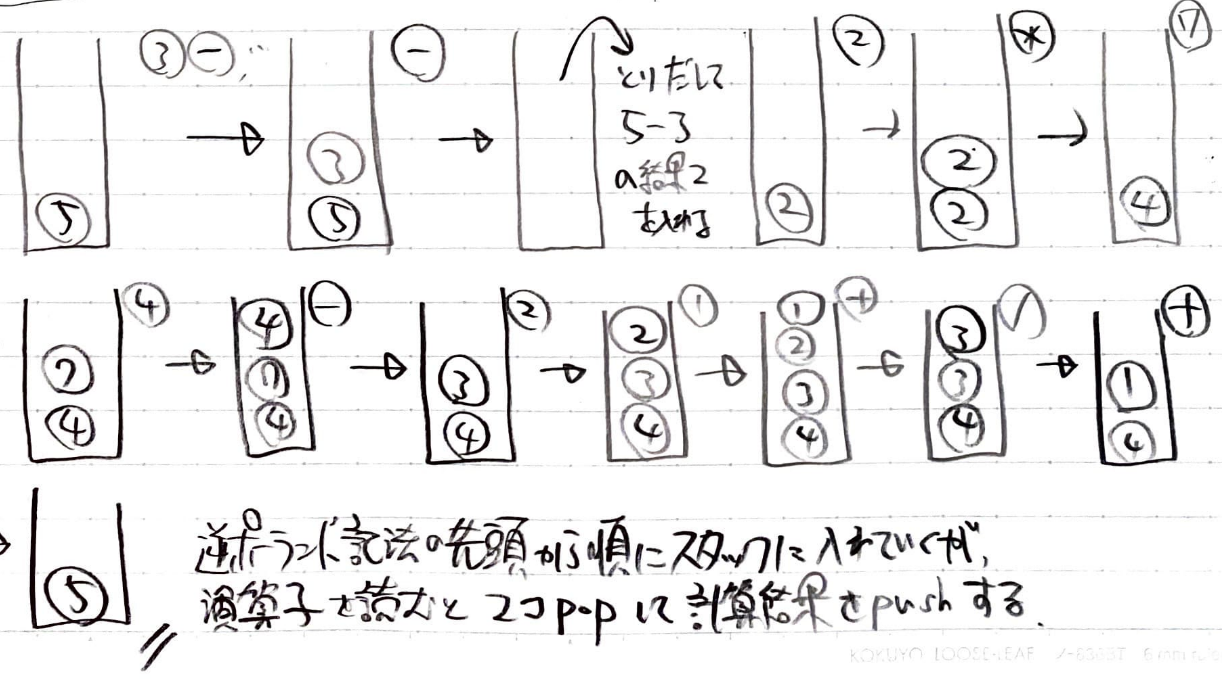
```
dfs(tree, root){
    if(rootノードに子がいる)
        dfs(tree, root.left)
        dfs(tree, root.right)
    end if
    rootのラベル出力する
}
dfs(構文木, 根の"+")
```

一般の深さ優先探索dfs

```
dfs(graph, v)
  ① seen[v] ← true
  ② for (next_v : G[v])
       if (seen[next_v]) continue
       dfs(graph, next_v)
```

再帰にするか // 訪問予定集合todo作るか

(4)



逆ポーランド記法の先頭から順にスタックに入れていくが、
演算子を読むと2コpopして計算結果をpushする。

3 $A=\{a_1,\dots,a_n\}$, $s\geq 0$. $d(i,j)=\{A_i' \subset A_i$ の数 $\mid \sum_{A_i'} a = j\}$

(1) $A=\{10,3,6,13,11,4\}$ のとき

$A_4=\{10,3,6,13\}$。この部分集合で和が16となる A_4' は

$A_4'=\{10,6\},\{3,13\}$ より $d(4,16)=2$.

$A_6=A$ の部分集合で和が20となる A_6' は

$A_6'=\{10,6,4\},\{3,6,11\},\{3,13,4\}$ より $d(6,20)=3$

(2) ?

$$d(i,j)=d(i-1,j-a_i)+d(i-1,j)$$

i個の部分和でjを作る $=\begin{cases} \text{i個目を使うか} = d(i-1,j-a_i) \\ \text{使わないか} = d(i-1,j) \end{cases}$

$j-a_i<0$ すなわち $a_i>j$ のとき $d(i,j)=d(i-1,j)$ (a_i は必ず使わない)

以上まとめると $d(i,j)=\begin{cases} d(i-1,j-a_i)+d(i-1,j) & (a_i \leq j) \\ d(i-1,j) & (a_i > j) \end{cases}$

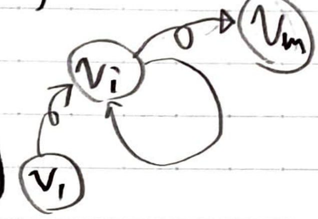
14 (1) $p=(4,2,1,3)$. $l(p)=5$.　　フロイド・ワーシャル法

(2) 最短経路 p が閉路を含むと仮定する。(背理法)

つまり

$$p=(v_1,\cdots,v_i,\cdots,v_j,\cdots,v_m)\quad(v_i=v_j,\ 1\leq i<j\leq m)$$

とする。このとき



$$l(p)=\sum_{k=1}^{m-1} l(v_k,v_{k+1})$$
$$=\sum_{k=1}^{i-1} l(v_k,v_{k+1})+\sum_{k=i}^{j-1} l(v_k,v_{k+1})+\sum_{k=j}^{m-1}(v_k,v_{k+1})$$

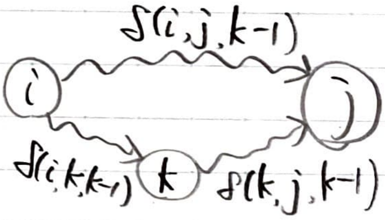
ここで $p'=(v_1,\cdots,v_i,v_{j+1},\cdots,v_m)$

とすると $l(p)$ の式で $(v_j,v_{j+1})=(v_i,v_{j+1})$ だから

$$l(p)=l(p')+\sum_{k=i}^{j-1} l(v_k,v_{k+1})$$
$$>l(p')$$

これは p が最短経路に矛盾。

(3) {1,2,…,k}の点を経由して i から j に到る経路は



(i) k を通らない。すなわち{1,…,k-1}を通る

(ii) k を通る。すなわち i から{1,…,k-1}を通って k に到り、k から{1,…,k-1}を通って j に到る。

の2通りに場合分けできる。

(i),(ii)の経路の和りはそれぞれ $\delta(i,j,k-1)$, $\delta(i,k,k-1)+\delta(k,j,k-1)$ と表され、$\delta(i,j,k)$ はこの2つの小さい方と一致するので題意は示された。 □

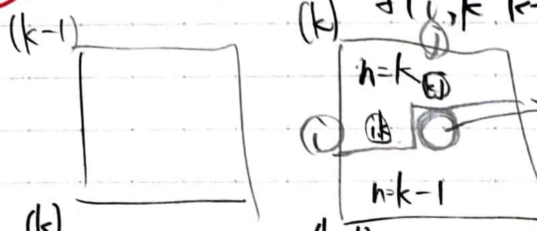
(4)

$$d^{(0)}=\begin{bmatrix}[0,&1,&2,&\infty],\\ [2,&0,&\infty,&\infty],\\ [\infty,&\infty,&0,&2],\\ [4,&1,&\infty,&0]\end{bmatrix}\qquad d^{(1)}=\begin{bmatrix}[0,&1,&2,&\infty]\\ [2,&0,&4,&\infty]\\ [\infty,&\infty,&0,&2]\\ [4,&1,&6,&0]\end{bmatrix}$$

(1,4)=(1,4,0) / (1,1,0)+(1,4,0)
2,1,0 + d(1,?,0)
4 1　1 3

$$d^{(2)}=\begin{bmatrix}[0,&1,&2,&\infty]\\ [2,&0,&4,&\infty],\\ [\infty,&\infty,&0,&2]\\ [3,&1,&5,&0]\end{bmatrix}\qquad d^{(3)}=\begin{bmatrix}[0,&1,&2,&4]\\ [2,&0,&4,&6]\\ [\infty,&\infty,&0,&2]\\ [3,&1,&5,&0]\end{bmatrix}\qquad d^{(4)}=\begin{bmatrix}[0,&1,&2,&4]\\ [2,&0,&4,&6]\\ [6,&3,&0,&2]\\ [3,&1,&5,&0]\end{bmatrix}$$

14 (5) $\delta(i,j,k) = \min\{\delta(i,j,k-1),$
$\delta(i,k,k-1) + \delta(k,j,k-1)\}$.



$\to d^{(k)}(i,j)$ で $d(i,k), d(k,j)$ 使うけど $d^{(k)}(i,k)\ d^{(k)}(k,j)$ で依存!
→ $d^{(k)} = d^{(k-1)}$ だから大丈夫だと示せばよい
→ 前から変わってない.

$$d^{(k)}(i,j) = \min\{d^{(k-1)}(i,j),\ d^{(k-1)}(i,k) + d^{(k-1)}(k,j)\}$$

$d^{(k)}(i,j)$ を計算するときは,
元に格納されていた値 $d^{(k-1)}(i,j)$ と
$d^{(k-1)}(i,k) + d^{(k-1)}(k,j)$ を比べる.

ここで
$$d^{(k)}(i,k) = \min\{d^{(k-1)}(i,k),\ d^{(k-1)}(i,k) + d^{(k-1)}(k,k)\} = d^{(k-1)}(i,k)$$
同様に $d^{(k)}(k,j) = d^{(k-1)}(k,j)$
だから $d^{(k)}(i,j)$ を計算するときは 1つの2次元配列の値のみ使えば十分.
つまり $d^{(k)}(i,j) = \min\{d^{(k-1)}(i,j),\ d^{(k)}(i,k) + d^{(k)}(k,j)\}$ であるから 1つでも大丈夫.

(c) N×N の d を ∞ で初期化.

```
for i ∈ V
    d[i][i] ← 0
for (i,j) ∈ E
    d[i][j] ← l(i,j)
for k ∈ {1, ..., N}
    for i ∈ {1, ..., N}
        for j ∈ {1, ..., N}
            if d[i][k] + d[k][j] < d[i][j] then
                d[i][j] ← d[i][k] + d[k][j]
return d
```