ワーシャル・フロイド法（全点対最短路問題）

蟻本 p.98 より

すべての 2 点間の最短路を求める問題を全点対最短路問題という。

頂点 0 ～ k と i, j のみを使う場合の、i から j への最短路を $d_{k+1}[i][j]$ とする。k = -1 のときは、i, j のみを使うとして $d_0[i][j] = \mathrm{cost}[i][j]$

頂点 0 から k のみを使う場合を、頂点 0 から k-1 のみを使う場合に帰着させる。頂点 0 から k のみを使うとき、i から j への最短路は、頂点 k を 1 回だけ通る場合と、まったく通らない場合のいずれかに分類できる。頂点 k を通らない場合は、$d_k[i][j] = d_{k-1}[i][j]$ となる。頂点 k を通る場合、経路は i から k までと k から j までに分解されるので、$d_k[i][j] = d_{k-1}[i][k] + d_{k-1}[k][j]$ となる。この二つをあわせて、$d_k[i][j] = \min(d_{k-1}[i][j], d_{k-1}[i][k] + d_{k-1}[k][j])$ となる。

これを用いて、同じ配列を使いまわして d[i][j] = min(d[i][j], d[i][k]+d[k][j] と更新していくアルゴリズムをワーシャル・フロイド法といい、$O(|V|^3)$ で全点対最短路を求めることができる。

```
     public static void WarshallFloyd(int[,] d, int v)
    {
for (int k = 0; k < v; k++)
    for (int i = 0; i < v; i++)
for (int j = 0; j < v; j++)
    d[i, j] = Math.Min(d[i, j], d[i, k] + d[k, j]);
    }
```