第 6 章　問題の解き方

文責：大和

必須学習項目：1(1)(2)

要望学習項目：教科書 6.2.1(a)

個別問題範囲：全部

1.　アルゴリズム

(1)　アルゴリズムの役割

・「コンピュータに問題を解かせる」とは

…問題をモデル化し、モデル化された問題の解を計算するプログラムを作り、それをコンピュータにやらせることである。

・具体的な問題からプログラムを作る過程

① 問題をモデル化する。

② モデル化された問題に対して、それをとく計算手順を考える。

③ 手順どおりに計算するプログラムを作る。

の 3 段階に分かれている。ここでプログラム化する前の段階の計算手順を**アルゴリズム**と呼ぶ。

(2)　アルゴリズムの実例

$\sqrt{x}$ の近似値を精度 δ で求める($\left|\sqrt{x}-y\right| < \delta$ となるような y を 1 つ求める)

(a)　反復法

小さい値から 0, δ,2δ,3δ,….と計算し $\sqrt{x}$ の近似値を求める方法。

実際に書くと

y←0　　　…y は 0 から始める。($\sqrt{x} > 0$ だから)

while $(y+\delta)^2 < x$ **do**

　　y←y+δ

done

return y　　　…y を解として計算を終了するという意味

この方法は、δの値に逆比例して計算回数が増える。

効率化された方法として、1 の位、0.1 の位、・・・と順に求めていく方法

→解の範囲に注目して、それを狭めていく方法→二分法

(b)　二分法…解の存在範囲を二分し、それを狭めていく

a ← 0

b ← x

while $b-a > \delta$ **do**

　　c ← (a+b)/2

　　if $c^2 > x$ **then** b ← c **else** a ← c **endif**

done

return a

[解釈]

$\sqrt{x}$ は区間 [a,b] に存在する。(初区間は[0,x])

c を a,b の中点とする。

x がそれより小さいなら区間の上限が c、そうでないなら下限が c

計算の過程は、p.123,124 表の通り。重要なのは、反復法が問題を解くために 14142 回かかったのに対し、この計算方法なら 15 回で収まる。このようにアルゴリズムによって計算回数は大幅に変化する。(→レジュメ p.10)

(3) 最短経路問題

点集合 V={$v_1, v_2, \ldots, v_n$}

v_1を始点として各点 v_k への最短距離 d_k (k=2,3,…n)を求める。

ネットワークが有向閉路(エッジの向きに進んだときに、ぐるぐると回れてしまうような閉路)を持たないとき簡単に解ける。

このとき点の添え字を適当に変更すれば、辺の向きが小さい添え字から大きい添え字へ向かうようにできる。そして、隣り合う 2 点 v,w の距離を l(v,w)とすると、

$d_j = \min\{d_k + l(v_k, v_j) \mid k$ はv_kとv_jが繋がっている k}

という式に j=2,3,…と順に代入することで、解ける。

例えば下の例だと

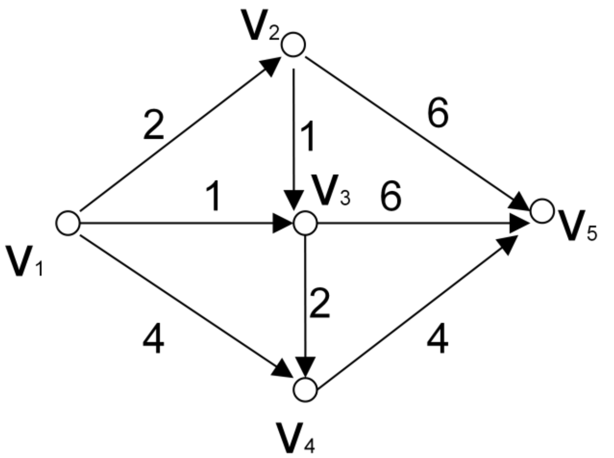
$d_2 = \min\{d_k + l(v_k, v_2) \mid k = 1) = 2$

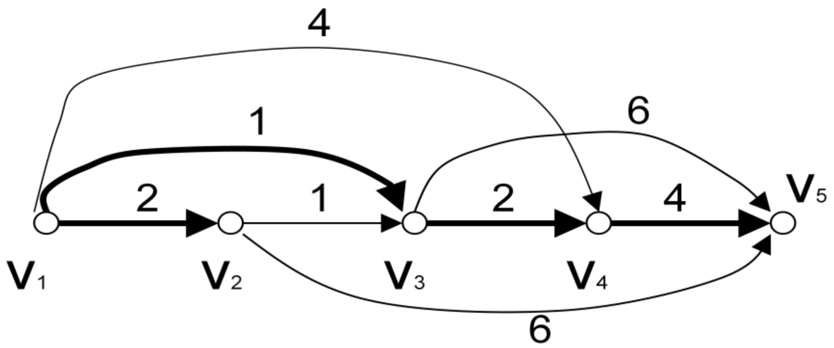
$d_3 = \min\{d_k + l(v_k, v_3) \mid k = 1,2) = \min\{d_1 + l(v_1, v_3), d_2 + l(v_2, v_3)\} = \min\{1,3\} = 1$

$d_4 = \min\{d_k + l(v_k, v_4) \mid k = 1,3) = \min\{d_1 + l(v_1, v_4), d_3 + l(v_3, v_4)\} = \min\{4,3\} = 4$

$d_5 = \min\{d_k + l(v_k, v_5) \mid k = 2,3,4) = \min\{d_2 + l(v_2, v_5), d_3 + l(v_3, v_5), d_4 + l(v_4, v_5)\} =$ $\min\{8,7,7\} = 7$

つまり$v_1 \rightarrow v_3 \rightarrow v_5$ と $v_1 \rightarrow v_3 \rightarrow v_4 \rightarrow v_5$ が最短距離ってことです。





点をこのように並び替えると、左から最短距離が求まっていく様子が分かりやすい。

おそらくv_1からv_5の最短距離を表しているであろう太い辺を間違えたのは中村です。私じゃありません。

(4) ハノイの塔問題

問題はレジュメ p.12 参照。解答は p.13 参照。

おそらく指数関数の増え方はすごいぞー、ってことが言いたいのだろう。(→p.14)

(5) 計算量　　　　　　　　(教科書 p.134)

計算量とは、アルゴリズムをもとにしたプログラムの実行時間を見積もるための指標。その見積もりは**計算量のオーダー**と呼ばれる非常に大まかな尺度である。

ex.)(2)の問題だと、(a)反復法の場合、$\sqrt{x}/\delta$ (b)二分法の場合、$\log_2(x/\delta)$ (→教科書 p.135)

(a)の方が非効率。

良いアルゴリズムとは、計算量のオーダーが n の多項式で押さえられるものである。

(cf.数３の極限で習うが、発散の早さは、「指数関数 ≫ 多項式 ≫ 対数関数」

P 問題…多項式(…polynomial)時間のアルゴリズムで解ける問題

NP 問題…その問題に対する答えが与えられたとき、その答えが正しいかどうかを多項式時間で調べられるような問題　　　　(NP=Not P ではない)

[雑学]

全ての P 問題は、同時に NP 問題でもあることが知られている。

逆に全ての NP 問題が P 問題であるか否かは未だに不明であり、「P≠NP 予想」として現代数学上の未解決問題の中でも最も重要な問題の一つとされ、2000 年にクレイ数学研究所のミレニアム懸賞問題の一つとして、この問題に対して 100 万ドルの懸賞金がかけられている。誰か解決して酒でもおごってください。[雑学終了]

NP 完全問題…NP に属する問題で、もしその問題に多項式時間の解法が存在すれば、NP に属する全ての問題が多項式時間のアルゴリズムで解ける、という性質を持つ問題

(6)　有限状態機械

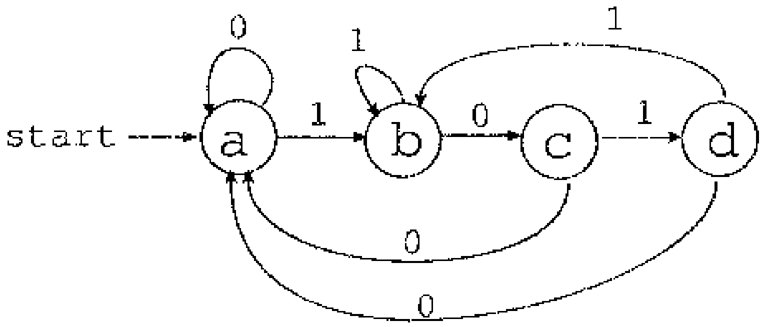
ゴーショーの神プリを参照。

オートマトンは絶対に出るそうです。

図を解釈させる場合と図を描かせる場合がありますが、まぁ気合いで乗り切ってください。

<u>おまけ</u>

過去問：記号集合｛0，1｝からの記号列を受けつけるオートマトンの状態遷移図が以下に示されている。このオートマトンの初期状態は a で、受理集合は F={d}であるとする。このとき受理される記号列の全体はどのような集合になるか述べよ。(2007 バカ村)



解答：(間違ってても知りません)

・文字列が「11」及び「00」を含まない場合、

「1」のあと「01」を奇数回繰り返したもの、もしくは「01」を偶数回繰り返したもの。

・文字列が「11」「00」を含む場合、

そのうち最後に出現するのが「11」の場合、そのあと、「01」を奇数回繰り返したもの。

最後に出現するのが「00」の場合、そのあと「1」、続いて「01」を奇数回繰り返したもの。

解説：

「00」がでると a、「11」が出ると b へと必ず行くことを利用した解答。

「00」「11」を含まないと「10101…」または「01010…」と繰り返しになるので考えやすい

しかし、こんな煩雑な解答になるはず無いので、自分で考えてください。

注意：

「末尾が 00101」とか「末尾が 1101」とかは合ってるけど必要十分じゃない気がする。(「末尾が 11010101」とかもある。かといって「末尾が 101」じゃ不十分)

「末尾が～」系で必要十分な解答あるかもしれませんが私の能力不足で分かりません。

あと、この解答を試験時間中に作ることは不可能です。