



アルゴリズムとデータ構造III

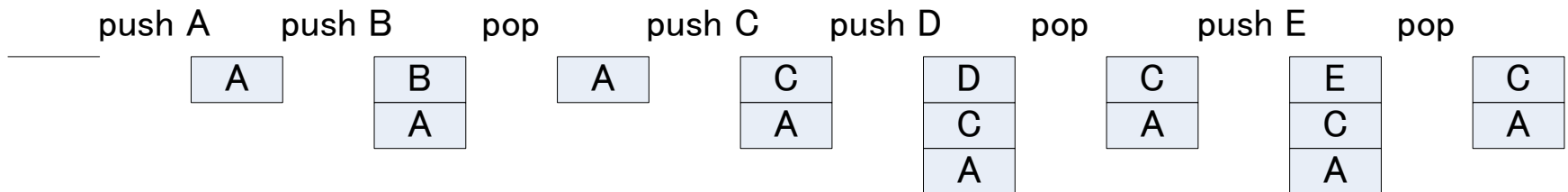
10回目: 12月10日

中間試験

授業資料 <http://ir.cs.yamanashi.ac.jp/~ysuzuki/algorithm3/index.html>

問題1. (スタック 平成21年秋期 基本情報技術者 午前 問5より)

- 空のスタックに対して次の操作を行った場合、スタックに残っているデータはどれか。ここで、“push x”はスタックへデータ x を格納し、“pop”はスタックからデータを取り出す操作を表す。
- push A → push B → pop → push C → push D → pop → push E → pop
- 解答例: AとC



問題2. (文脈自由文法)

- 文脈自由文法と文脈依存文法の違いを200文字以内で説明せよ.
- 文脈依存文法の生成規則は $u\alpha v \rightarrow u\beta v$ (α は非終端記号, β, u, v は終端または非終端記号列) の形で表される. これは非終端記号 α の前後の記号 u, v により α から β の導出が制限される事を意味する.
 $u\alpha v \rightarrow u\beta v$ で u と v が ε であるとき $\alpha \rightarrow \beta$ となり, 前後の記号(文脈)は α から β の導出に影響を与えない. そのため $\alpha \rightarrow \beta$ のような生成規則だけを持つ文法を前後の文脈(記号)に対して影響を受けないという意味で文脈自由文法と呼ぶ.



問題3. (構文解析)

- 構文解析の代表的手法を3つ挙げよ.
- 解答例:
 - CYK法
 - Earley法(アーリー法)
 - トップダウンチャート法
 - LR法など

問題4. (CYK法)

- 下の図は「Nana met with Ramos.」を構文解析中のCYK表である。
- 1) 図中の①, ②, ③, ④, ⑤, ⑥には何が入るか答えよ。
- 2) CYK表から得られる「Nana met with Ramos.」の構文木を描け。

	Nana	met	with	Ramos
Nana	N → Nana	①	④	⑥
met		V → met	②	⑤
with			P → with	③
Ramos				N → Ramos

書き換え規則

$S \rightarrow N VP$

$S \rightarrow N V$

$VP \rightarrow VP PP$

$VP \rightarrow V PP$

$PP \rightarrow P N$

$N \rightarrow Nana$

$N \rightarrow Ramos$

$V \rightarrow met$

$P \rightarrow with$

①: $S \rightarrow N V$

②: 該当なし

③: $PP \rightarrow P N$

④: 該当なし

⑤: $VP \rightarrow V PP$

⑥: $S \rightarrow N VP$

問題4. (CYK法)

- 下の図は「Nana met with Ramos.」を構文解析中のCYK表である。
- 1) 図中の①, ②, ③, ④, ⑤, ⑥には何が入るか答えよ。
- 2) CYK表から得られる「Nana met with Ramos.」の構文木を描け。

	Nana	met	with	Ramos
Nana	N → Nana	①	④	⑥
met		V → met	②	⑤
with			P → with	③
Ramos				N → Ramos

書き換え規則

$S \rightarrow N VP$

$S \rightarrow N V$

$VP \rightarrow VP PP$

$VP \rightarrow V PP$

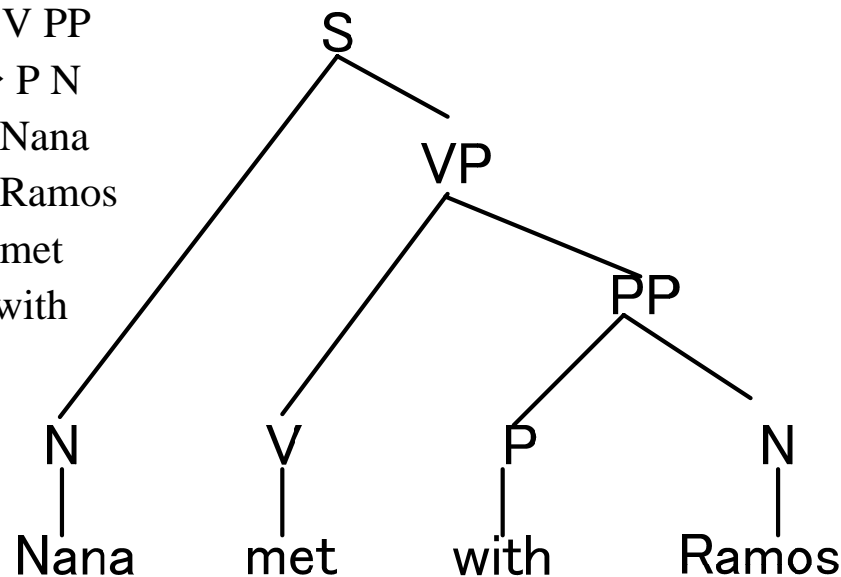
$PP \rightarrow P N$

$N \rightarrow Nana$

$N \rightarrow Ramos$

$V \rightarrow met$

$P \rightarrow with$





問題5. (トップダウンチャート法)

- CYK法と比較したときのトップダウンチャート法の特徴を簡潔に説明せよ.
- 文脈自由文法で書かれた文を構文解析するための代表的な手法
- アークとノードを使ったグラフで表される
- CKY法ではチョムスキーの標準形以外は扱えないが、チャート法では $X \rightarrow ABC$ のような変換規則も扱うことができる.
- 簡単な予測を使うことが出来るため、CKY法より効率がよい



問題6. (動的計画法)

- 動的計画法を200文字以内で説明せよ.
- 解くのに時間のかかる問題を、複数の部分問題に分割することで効率的に解くアルゴリズム
- 動的計画法の適用例として、最短経路検索のためのダイクストラ法, パターンマッチングのためのDPマッチングがある.

問題7. (ダイクストラ法 平成15年 秋期 基本情報技術者 午後 問04より)

■ 問題は長いので省略

■ 解答例:

- a: キ $Z = D[Y]$
- b: エ $S[Y] = T$
- c: カ $Y \leftarrow S[Y]$
- d: ア $X > 0$

```
[プログラム] プログラム名: SP(N, C[, ])
実数型: C[, ], D[N], Z
整数型: P[N], S[N], W[N], N, T, X, Y
(初期設定)
X ← 1
while (X ≤ N){
    D[X] ← C[1, X]
    P[X] ← 0
    S[X] ← 1
    X ← X+1
}
P[1] ← 1
```

(最短経路を求める処理)

```
X ← 2
while (X ≤ N){
    Y ← 2
    Z ← ∞
    while (Y ≤ N){
        if ((P[Y] = 0) and (D[Y] < Z)){
            T ← Y
            a
        }

        Y ← Y+1
    }
    P[T] ← 1
    Y ← 2
    while (Y ≤ N){
        if ((P[Y]=0) and (D[Y] >
(D[T]+C[T,Y]))){
            D[Y] ← D[T] + C[T,Y]
            b
        }
        Y ← Y+1
    }
    X ← X+1
}
```

(最短経路の出力処理)

```
X ← 1
Y ← N
while (Y ≠ 1){
    W[X] ← Y
    c
    X ← X+1
}
W[X] ← Y
while (d){
    Output(W[X])
    X ← X - 1
}
```

問題8. (DPマッチング)

- 下の表は「abcd」と「accd」の単語間距離をDPマッチングにより計算しているところである. 表中の①, ②, ③, ④, ⑤には何が入るか答えよ. 但し, 不一致ペナルティは3点, 挿入ペナルティ=1, 脱落ペナルティ=1とする.

通行ペナルティ積算表

	a	b	c	d
a	0	4	8	12
c	4	3	4	①
c	8	7	②	③
d	12	11	④	⑤

① : 8

② : 3

③ : 7

④ : 7

⑤ : 3



問題9. (A*アルゴリズム)

- A*アルゴリズムとダイクストラ法の類似点と相違点を300文字以内で説明せよ
- 類似点
 - どちらも最短経路問題を解くのに使われる.
 - マイナスのコストをもつ辺を含む経路の最短経路は解くことが出来ない.
- 相違点
 - ダイクストラ法はスタートから各節点までの移動コストを利用して最短経路を求めるのに対してA*アルゴリズムはスタートから各節点までの移動コストと節点からゴールまでの予測コスト($0 \leq \text{予測コスト} \leq \text{実際のコスト}$)を利用する.
 - A*アルゴリズムは予測コストを利用するためダイクストラ法よりも効率よく問題を解くことが出来る. 各節点からゴールまでの予測コストがすべて0である場合, ダイクストラ法のアルゴリズムと同じになる.

問題10. (平成19年 春期 基本情報技術者 午前 問78より)

- 図中の矢印に記した数値は、各区間の運賃を表す。出発地から目的地までの経路のうち、最も安い総運賃はいくらか。また、その時の経路を示せ。

- 解答例:

- 最も安い総運賃: 20
- そのときの経路: 出発地 → B地点 → F地点 → H地点 → 目的地

