

# 言語処理工学 A 期末テスト 2015/2/6 井上克郎

教科書、ノート、その他持ち込み なし！ [1]は解答用紙表紙に、[2]は内側の2ページに、[3]は裏表紙に書け。

[1] 以下のアセンブリコード A を実行コストを下げるために B に書き換えた。教科書で用いられた実行コストモデルを使って B を使うメリットを定量的に議論せよ。

コード A

```
L6: MOV  V, R4
      ADD  #1, R4
      MOV  R4, V
```

コード B

```
L6: INC  V
```

[2] 次の3番地コードに関して答えよ。(解答用紙2-3ページに答えを書くこと)

```
1  a=1
2  b=2
3 L1: if a<b goto L2
4    goto L3
5 L2: a=a+1
6 L3: b=b×2
7    if a>b goto L1
```

(2-1) このプログラムの基本ブロックに分け、各ブロックにプログラムの上から順に、番号 B1~ をつけ、各ブロックに入るコードを行番号で明示せよ。

(2-2) フローグラフを書け。

(2-3) 得られたフローグラフの支配木 (dominator tree) を書け。この木の各頂点はブロックの番号とせよ。

(2-4) フローグラフ中の全バックエッジを挙げ、それぞれが構成する Natural Loop の各頂点を示せ。

[3] 先の[2]の3番地コードに関して答えよ。(解答用紙4ページ目に書くこと)

(3-1) 各基本ブロックの GEN と KILL の集合を求めよ

(3-2) データフロー方程式を立てて、各基本ブロックの IN と OUT 集合を求めよ。各集合の初期値から順に変化して停止するまでの各ステップを書け。

[1] 30点  
 2-1-A  
 L6: MOV V, RY  
 ADD #1, RY  
 MOV RY, V

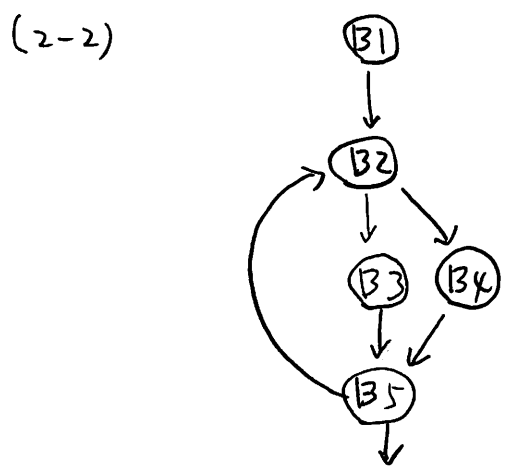
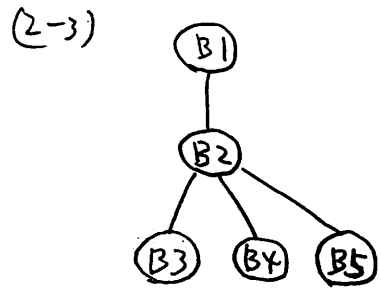
1 word 分 2 2 1  
 命令 2 word, 実行 V が 3 CPU 1 word. 計 3  
 " 2, 実行 0 計 2  
 " 2, CPU が 3 V 1 word. 計 3

総計 8

2-1-B  
 L6: INC V 命令 2, V → CPU → V 2 回 計 4

よ、て B に対応により、A の半分 の コスト で 実行 できる。

[2] 10点 x 4 (2-1) B1 [1] — leader  
 [2]  
 B2 [3] — leader  
 B3 [4] — leader  
 B4 [5] — leader  
 B5 [6] — leader  
 [7]



(2-4)  
 バックエッジ  
 B5 → B2  
 このバックエッジに因る Natural Loop  
 {B2, B3, B4, B5}

[3]

(3-1)

10分

	Gen	Kill	又	Gen	Kill
	1 2 3 4 5 6 7	1 2 3 4 5 6 7			
B1	1 1 0 — 0	0 — 0 1 1 0		{1, 2}	{5, 6}
B2	0 — 0	0 — 0		∅	∅
B3	0 — 0	0 — 0		∅	∅
B4	0 0 0 0 1 0 0	1 0 — 0		{5}	{1}
B5	0 — 0 1 0	0 1 0 — 0		{6}	{2}

(3-2)

20分

	IN[Bi]	OUT[Bi]	
B1	0 — 0	1 1 0 — 0	Initial
B2	"	0 — 0	
B3	"	"	
B4	"	0 0 0 0 1 0 0	
B5	"	0 — 0 1 0	

B1	0 — 0	1 1 0 — 0	Step 1
B2	1 1 0 0 0 1 0	1 1 0 0 0 1 0	
B3	0 — 0	0 — 0	
B4	0 — 0	0 0 0 0 1 0 0	
B5	0 — 0 1 0 0	0 — 0 1 1 0	

B1	0 — 0	1 1 0 — 0	Step 2
B2	1 1 0 0 1 1 0	1 1 0 0 1 1 0	
B3	1 1 0 0 0 1 0	1 1 0 0 0 1 0	
B4	1 1 0 0 0 1 0	0 1 0 0 1 1 0	
B5	0 0 0 0 1 0 0	0 0 0 0 1 1 0	

B1	0 — 0	1 1 0 — 0	Step 3
B2	1 1 0 0 1 1 0	1 1 0 0 1 1 0	
B3	1 1 0 0 1 1 0	1 1 0 0 1 1 0	
B4	1 1 0 0 1 1 0	0 1 0 0 1 1 0	
B5	1 1 0 0 1 1 0	1 0 0 0 1 1 0	

B1	0 — 0	1 1 0 — 0	Step 4 又 Step 3 同 5, 2 収束 → halt
B2	1 1 0 0 1 1 0	1 1 0 0 1 1 0	
B3	1 1 0 0 1 1 0	1 1 0 0 1 1 0	
B4	1 1 0 0 1 1 0	0 1 0 0 1 1 0	
B5	1 1 0 0 1 1 0	1 0 0 0 1 1 0	