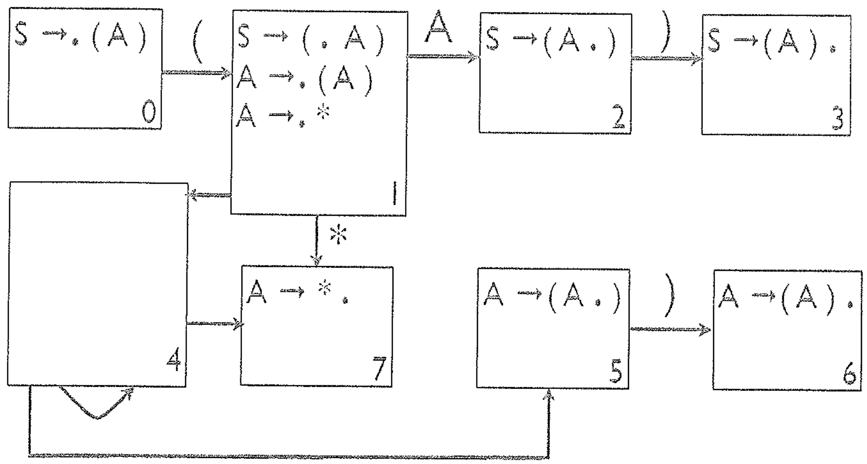
CE-2
August 2012 QE

Question 1 (30 points total): Given the grammar G:

S → (A)
A → (A)
A → *



(a) (12 points) Complete the LR(0) CFSM in your blue book. You only need to show state 4 and labels on edges adjacent to it in your solution.

(b) (12 points) Given the parse sequence below, and your table above

1. What action occurs going from step 6 to 7
2. What action occurs going from set 7 to 8?
3. At 11, is the grammar accepted or in an error state?

For actions, give accept, shift or reduce, and the symbol shifted or production reduced.

step	state stack symbol stack	step	state stack symbol stack	step	state stack symbol stack
1	0 λ	5	0 1 4 4 7 (((*	9	0 1 4 5 6 ((A)
2	0 1 (	6	0 1 4 4 5 (((A	10	0 1 2 (A
3	0 1 4 ((	7	0 1 4 4 5 6 (((A)	11	0 1 3 (A)
4	0 1 4 4 (((	8	0 1 4 5 ((A		

(c): (6 points) Given a state stack containing "0 1 4 4 4 4 4 7"

i. How many "(" are in the string?
ii. How many ")" are in the string (after the string is accepted by the parser)?

Write in Exam Book Only

Question 2: (24 points total) Given the two loops:

```
S1: for (j = 0; j < n; j++) {          S7:  for (k = 0; k < n; k++) {
S2:    for (k = 0; k < n; k++) {       S8:     for (j = 0; j < n; j++) {
S3:       a[j][k] = . . .;             S9:        a[j][k] = . . .;
S4:        . . .  = a[j-1][k+1];       S10:        . . .  = a[j-1][k+1];
S5:    }                               S11:    }
S6: }                                  S12: }
```

and an "a" array that is too large to fit in cache laid out in memory as:

1	a[0][0]	a[0][1]	. . .	a[0][n-1]	a[1][0]	a[1][1]	. . .	a[1][n-1]	. . .	a[n-1][0]	a[n-1][1]	. . .	a[n-1][n-1]

and an "a" array that is too large to fit in cache laid out in memory as:

2	a[0][0]	a[1][0]	. . .	a[n-1][0]	a[0][1]	a[1][1]	. . .	a[n-1][1]	. . .	a[0][n-1]	a[1][n-1]	. . .	a[n-1][n-1]

i. **(5 points) For the left loop (the *j* loop is outermost) which layout is best (put the letter of the correct answer in your blue book):**
   a. layout 1
   b. layout 2
   c. no difference between the two

ii. **(5 points) For the right loop (the *k* loop is outermost) which layout is best (put the letter of the correct answer in your blue book):**
   a. layout 1
   b. layout 2
   c. no difference between the two

iii. **(7 points) Give the direction and type of dependence, if any, on the write of "a" in S3 and the read of "a" in S4.**

iv. **(7 points) Give the direction and type of dependence, if any, on the write of "a" in S9 and the read of "a" in S10.**

Write in Exam Book Only

Question 3 (18 points): Given the loop:

```
for (i = 2; i < 4*n+2; i++) {
    a[i][j] = a[i-1] + a[i-2];
}
```

that is unrolled to form the loop

```
for (i = 2; i < 4*n+2; i+=4) {
    a[i][j] = a[i-1] + a[i-2];
    a[i+1][j+1] = a[i] + a[i-1];
    a[i+2][j+2] = a[i+1] + a[i];
    a[i+3][j+3] = a[i+2] + a[i+1];
}
```

Write the letters of all that are true in your blue books.

a. More opportunities exist after instruction level parallelism after unrolling

b. Fewer opportunities exist after instruction level parallelism after unrolling

c. There is no affect on instruction level parallelism or the effect is unknowable

d. Data cache performance is worse after unrolling

e. Data cache performance is better after unrolling

f. There is no effect on data cache performance or the effect is unknown

g. More registers are needed by a register allocator doing local (per-basic block) register allocation

h. Fewer registers are needed by a register allocator doing local (per-basic block) register allocation

i. There is no effect on the number of registers are needed by a register allocator doing local (per-basic block) register allocation or the effect is unknowable.

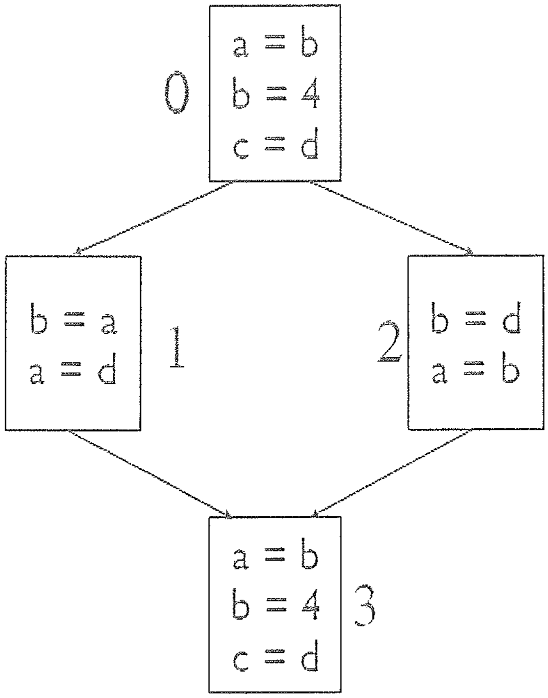
j. The loop could be parallelized before unrolling, but cannot be parallelized after unrolling

k. The loop could not be parallelized before unrolling, but can be after unrolling

l. The unrolling of the loop has not effect on parallelization.

Question 4 (28 points): For this question you need to perform a live variable analysis.

i. (3 points) What goes in the GEN set?
ii. (3 points) What goes in the KILL set?
iii. (3 points) Is the analysis an ANY or ALL paths algorithm?
iv. (3 points) Is the analysis a FORWARD or BACKWARDS algorithm?
v. (16 points) Given the control flow graph of a program below, fill a table like the one below in your blue books.



BLOCK	IN	OUT	GEN	KILL
0				
1				
2				
3				

Write in Exam Book Only