Department of Computer Science and Engineering

Indian Institute of Technology, Bombay

MIDSEMESTER EXAMINATION, 2004-2005 SESSION

CS 616 : PARALLELIZING COMPILERS

Date : 17-09-2004 **Day : Friday**
Time : 14:30 hrs to 16:30 hrs **Marks : 50**
Weightage : 30%

Remarks : 1. Open Notes Examination (no books/xerox copies).
2. Exchange of material is not permitted.
3. Please answer the questions in the order given.

Q1. Consider the loop given below :

```
     do I = 1 , 10
S :       a[3*I - 2] = ...
T :       ... = a[2*I + 2 ] + 25
     endo
```

(a) Find all source-sink pairs between S and T using the *exact test.*
(b) Write all dependence information between S and T using the solutions obtained in part(a) above. Indicate for each dependence, the type of dependence, distance and direction vector as appropriate and draw the data dependence graph. (8 + 2 = 10 marks)

Q2. Consider the loop given below :

do $I_1 = 1$, 50
  do $I_2 = 1$, 50
S: a[$I_1 - I_2 + 10$] = ...
T: ... = a[$-2 * I_1 + I_2 + 157$]
  endo
endo

(a) It is required to compute the bounds using Banerjee's inequalities under some specified direction vectors. Complete the entries against the bounds of the following table and show the steps leading to the bounds.

direction vector	lower bound, LB	Upper bound, UB
$(<,*)$	[,]	[,]
$(=,*)$	[,]	[,]
$(<,>)$	[,]	[,]

(b) Using the bounds obtained above, interpret them to report the data dependences between S and T. (6 + 2 = 8 marks)

Q3(a). Prove that the parallelism index of a statement S, denoted by $\rho(S)$, and defined by $\rho(S) = m - \eta(S,S)$, is always ≥ 0. The term $\eta(S,S)$ denotes the nesting level of dependence from statement S to itself and m is the number of loops containing S.

(b). In ACK's method for parallel code generation, why was the procedure check_sons() rewritten for the case of a serial region having deeper parallelism ? Is it possible to have the number of barriers reduced by more than 1 by using the improved version of check_sons()? Justify your answer by constructing an appropriate example. (3 + 5 = 8 marks)

Q4. Examine the program fragment given below.

```
      do I = 1 , 10
         do J = 1 , 10
S1 :        x[ I ] = 5
            do K = 1 , 10
S2 :           a[ I, J, K ] = b[ I, J, K ] + 6
S3 :           b[ I + 1, J, K ] = a[ I, J, K ] + b[ I + 1, J, K - 1 ]
S4 :           c[ I + 1, J, K ] = d[ I, J, K ] + c[ I, J, K ]
S5 :           d[ I + 1, J + 1, K ] = c[ I + 1, J, K ] + a[ I, J, K ]
            endo
         endo
      endo
```

(a) Show the vector code produced by Allen-Kennedy's vector code generation Algorithm. Comment on the optimality of the code in the sense whether each statement is vectorized in the maximum number of loops enclosing it.

(b). Can the vector code of part (a) be improved by using loop interchange transformation ? Justify the loop interchanges that you wish to perform and show the final vector code for this program. (6 + 4 = 10 marks)

Q5(a). Show the parallel code generated using ACK's algorithm for the program fragment of Q4 above using the standard check_sons() procedure. Explain the presence of barrier points introduced in the parallel code.

(b). Would there be any improvement in the parallel code generated in part(a) above if the enhanced check_sons() is used instead ? Justify your answer by showing the parallel code produced in this context.

(c). Would loop-interchange transformation improve the quality of parallel code produced in part(a) or part(b) ? Indicate the loop interchanges that can be legally performed, if any, and show the improved parallel code.

(6 + 4 + 4 = 14 marks)

**** **End of the Paper** ****