**Q1. Write an in-mapper combiner algorithm modifying algorithm 3.8 (That is, pairs approach)**

class Mapper

Method Initialize

H = new AssociativeArray()

Method Map(docid a; doc d)

for all term w in doc d do

for all term u in Neighbors(w) do

H{pair(w;u)} ++

Method close

Emit(pair (w; u); H{pair(w;u)})

class Reducer

method Reduce(pair p; counts [c1; c2; …])

s = 0

for all count c in counts [c1; c2; …] do

s = s + c .

Emit(pair p; count s)

**Q2. Write an in-mapper combiner algorithm modifying algorithm 3.9 (That is, stripes approach)**

class Mapper

Method Initialize

H = new AssociativeArray()

Method Map(docid a; doc d)

for all term w in doc d do

for all term u in Neighbors(w) do

H{w}{u} = H{w}{u} + 1 . //Tally words co-occurring with w

Method close

Emit(Term w; Stripe H)

class Reducer

method Reduce(term w; stripes [H1;H2;H3; : : :])

Hf = new AssociativeArray

for all stripe H in stripes [H1;H2;H3; …] do

Sum(Hf; H) . //Element-wise sum

Emit(term w; stripe Hf )

**Q3**







