Queue := {all pairs i,j s.t. i != j}

BadTraces = {}

phaseGoodTraces = {}

While true:

While trace = queue.Pop; base != null:

If trace contains any badTrace from badTraces OR trace exists in phaseGoodTraces:  
 skip

If Gibbons(trace):

phaseGoodTraces .Add(trace)

Else:

BadTraces.Add(trace)

If phaseGoodTraces.Empty():

Break

Else:

Queue = {add a union for each pair from phaseGoodTraces}

phaseGoodTraces = {}

Return BadTraces