**Lab Solution Question Sets Answers**

**Lab 1**

Q1 Greedy → local optimum is the result of any one phase of the set of phases that compose the algorithm

Q2 We hope that the ‘local optimum’ equals the ‘global optimum’

Q3 Greedy → efficient and simple solution if an approximate answer is sufficient

Q4 File compression or Coin change

Q5 Traffic problems

**Lab 2**

Q1 Dynamic→ optimal solution is the best solution chosen or built from the sub solutions calculated for each substructure

Q2 true

Q3 No, the most efficient solution is often obtained by explicitly enumerating the distinct subproblems(substructure)

Q4 Yes

Q5 To allow the recursive code to be efficient

Q6 Persisting results of recursive call speed up a recursive algorithm by looking them up again later ← memoization

Q7 No, it is not necessary to alway retain all intermediate results

Q8 No for example we can’t perform arbitrary-precision arithmetic in constant time.0On different platforms and operating systems there will be different precision and overflow issues that will impact significantly on performance

**Lab 3**

Q1 The algorithm has at least 2 or more recursive calls, for example merge sort where the array is divided and the 2 recursive calls operate on the 2 halves of the array

Q2 The multi branch recursive calls have the potential to execute in parallel. The memoization structures can be cached

Q3 it is difficult to determine how long to keep memoized values → space resources can outweigh any CPU saving

Q4 Divide and conquer solutions can perform redundant calculations