Q, 4.2-1:

$$\begin{bmatrix} 1 & 3 \\ 7 & 5 \end{bmatrix} \cdot \begin{bmatrix} 6 & 8 \\ 4 & 2 \end{bmatrix} = \begin{bmatrix} 18 & 14 \\ 62 & 66 \end{bmatrix}.$$

$P_1 = 1(8-2) = 6$

$P_2 = (1+3) \cdot 2 = 8$

$P_3 = (7+5) \cdot 6 = 72$

$P_4 = 5(-2) = -10$

$P_5 = (1+5)(6+2) = 48$

$P_6 = (3-5)(4+2) = -12$

$P_7 = (1-7)(6+8) = -84$

4.2-2

```
MultiMatrix( A, B) {
    if ( A[0].size() == 1) {
        return A·B
    }
    else { n = A[0].size()
        A11 = A [0,...,n/2][0,...,n/2]
        A12 = A[0,...,n/2][n/2+1,...,n]
        A22 = A[n/2+1,...,n][n/2+1,...,n]
        A21 = A[n/2+1,...,n][0,...,n/2]
        B11 = B [0,...,n/2][0,...,n/2]
        B12 = B[0,...,n/2][n/2+1,...,n]
        B21 = B[n/2+1,...,n][0,...,n/2]
        B22 = B[n/2+1,...,n][n/2+1,...,n]
        P1 = MultiMatrix(A11, B12 - B22)
        P2 = MultiMatrix(A11 + A12, B22)
        P3 = MultiMatrix(A21 + A22, B11)
        P4 = MultiMatrix(A22, B21 - B11)
        P5 = MultiMatrix(A11 + A22, B11 + B22)
        P6 = MultiMatrix(A12 - A22, B21 + B22)
        P7 = MultiMatrix(A11 - A21, B11 + B12)
        C = new double [n][n]
        C[0,...,n/2][0,...,n/2] = P5 + P4 - P2 + P6
        C[0,...,n/2][n/2+1,...,n] = P1 + P2
        C[n/2+1,...,n][0,...,n/2] = P3 + P4
        C[n/2+1,...,n][n/2+1,...,n] = P5 + P1 - P3 - P7
        return C }
}
```

Scanned with CamScanner

Problem 2:

We can divide an array A with d parts. $d \le \frac{n}{2}$, where $n = A.size()$.
Because d is a fixed number, so, it needs constant time to compare in that recurrence.

$T(n) = d\,T(\frac{n}{d}) + O(1)$

$a = d,\ b = d,\ \log_b a = 1.$

By using master theorm,

$T(n) = \theta(n)$

Problem 3:

Greedy algorithm: we select the last activity to start. This means that we choice that looks best at the moment without regrad for future sequence.

Prove: set $S = \{1, 2, \cdots, n\}$ of n activities,
S_i, start time of activity i
f_i, finish time of activity i.

Assume $S_1 \le S_2 \le \cdots \le S_n$

1: There exists an optimal solution A such that the greedy choice "n" in A:

Prove: We order the activity in A by start time from lastest start to earliest start. Denote the first activity in A is k_n

If $k_n = n$, then A begins with a greedy choice.

if $k_n \ne n$, then $A' = (A - \{k_n\}) \cup \{n\}$

1. the sets $A - \{k_n\}$ and $\{n\}$ are disjoint
2. the activites in A' are compatible.
   This is because: $S_{k_n} \le S_n$
3. A' are also optimal, $|A'| = |A|$.

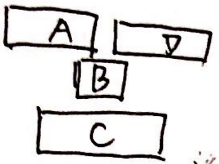
2. If A is an optimal solution which contains $\{n\}$, then $A - \{n\}$ is an optimal solution to $S' = \{i \in S,\ f_i \le S_n\}$

Proof: If there exists B' to S' such that $|B'| > |A'|$ then
$B = B' \cup \{n\}$
is the globally optimal solution, we have
$|B| > |A|$
which is contradicting to the optimal of A. so, B' does't exist.

Because of 1 and 2, we have greedy choice can yield optimal solution.

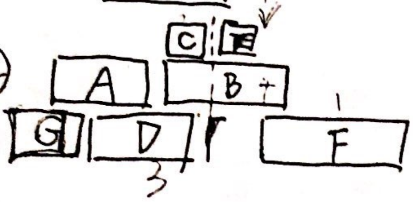
Scanned with CamScanner

Problem 4:

① 

optimal: 2 : A, D

select least duration: 1 : B

② 

optimal: 4. G D E F

select overlaps the fewest
and then select earliest time: 3 G E F