1. (10 points)

$N$ jobs are to be processed. Two machines $A$ and $B$ are available. If job $i$ processed on machines $A$, then $a_i$ units of processing time are needed. If it is processed on machine $B$, then $b_i$ units of processing time are needed. Because of the peculiarities of the jobs and the machines, it is quite possible that $a_i \geq b_i$ for some $i$ while $a_j < b_j$ for some $j, j \neq i$. Obtain a dynamic programming algorithm to determine the minimum time needed to process all the jobs. Note that jobs cannot be split between machines. State and prove time complexity of your algorithm.
2. (10 points)

Arbitrage is the use of discrepancies in currency exchange rates to make a profit. For example, there may be a small window of time during which 1 U.S dollar buys 0.75 British pounds, 1 British pound buys 2 Australian dollars, and 1 Australian dollar buys 0.70 U.S dollars. Then, a smart trader can trade one U.S dollar and end up with $0.75 \times 2 \times 0.7 = 1.05$ U.S dollars, a profit of 5

Suppose that there are $n$ currencies $C_1, \ldots, C_n$ and an $n \times n$ table $R$ of exchange rates, such that one unit of currency $C_i$ buys $R[i, j]$ units of currency $C_j$. Devise and analyze a dynamic programming algorithm to determine the maximum value of $R[C_1, C_i(1)] \cdot R[C_i(1), C_i(2)] \cdot \ldots \cdot R[C_i(k), C_1]$ where $i(1), \ldots, i(k)$ is a subset of $\{1, \ldots, n\}$. State and prove time complexity of your algorithm.