1. [20 pts] Reduce the following problems to 0-1 knapsack, or solve them from scratch using
dynamic programming:

(a) Given $N$ positive integers $x_1, x_2, \ldots, x_N$, find whether there exists a subset $S$ which has
sum exactly $Y$.

(b) Here, $\mathbb{Z}^+$ refers to the set of positive integers. Assume that $a_i, b_i$, and $c$ are also positive
integers which are known to you, and that $x_i$’s are the decision variables.

\[
\text{maximize } \prod_i x_i^{a_i} \\
\text{subject to:}
\quad (\forall i) : \quad x_i \in \mathbb{Z}^+, \\
\quad \text{and} \quad \sum_i b_i x_i \leq c
\]

For each problem analyze the running time of your solution and the state space of the un-
derlying dynamic program. Can you think of any real-life examples where either of the two
problems might arise? Implement either of these problems using your favorite programming
language and send us the code and some test inputs.

2. [10 pts] A total of of $n-1$ stepping stones were placed at one meter intervals across a river that
is $n$ meters wide. Over time, some of these stones got washed away. For $1 \leq i \leq n-1$, $A[i] = 1$
if the $i$-th stepping stone is still there, and $A[i] = 0$ if this stone got washed away. A baby
kangaroo wants to cross this river. The kangaroo can jump at most $k$ meters at a time.
Also, a jump gives a kangaroo some momentum; so two successive jumps can only differ by
one meter. For example, if the first jump is 10 meters, the next jump can be 9, 10, or 11
meters. Assume the kangaroo never jumps backwards. You are given $n, A, k$. You need to
find out whether the kangaroo can cross the river without falling into the water, and if yes, the
minimum number of leaps it needs to make. Assume that the kangaroo must start jumping
from the bank i.e. from the point $i = 0$ and is allowed to land anywhere on the opposite side
i.e. at any $i \geq n$. The kangaroo can take a running start, so the first jump can be anything
between 1 and $k$. Note: For a dynamic programming problem where you are just asked to
give a solution, you must clearly state the recursive formula, define the terms in the formula,
analyze the state space size and the running time, explain why the recursion does not lead to
cycles, and explain what decision variables you would store.

3. [10 pts] Suppose there are $n$ cities along the Amazon river, at positions $x_1, x_2, \ldots, x_n$. We have
to choose $p$ of these cities as docks such that the sum, over all cities, of the distance of the city
to the nearest *downstream* dock is minimized. Present a dynamic programming algorithm for this problem.

4. **[15 pts]** A marketing executive has an advertising budget of $B$ dollars, which needs to be allocated to $n$ products. The expected return from investing $x_i$ dollars in advertising product $i$ is $f_i(x_i)$ where the function $f_i$ is given either as a table or a formula. The executive wants to maximize total expected return.

(a) Assume that each $f_i$ is concave. Reduce this problem to a min-cost flow problem.

(b) In reality, $f_i$ would often be non-concave for this problem. Explain why (this requires a common-sense answer, not a technical one).

(c) Explain how you could solve this problem using a dynamic program, assuming that the $f_i$’s are completely general.