EECS 336 Fall 2015
Homework Problem 6.3

Define $s_i = d_i - f_i$
Step One: $OPT(i, j)$ = "set of all possible sums using j elements from $\{s_1, ..., s_i\}$"

Step Two: Recurrence. $OPT(i+1, j+1) = OPT(i, j+1) \cup [OPT(i, j) + s_{i+1}]$

Step Three: Base case
$OPT(i, i) = \sum_{l=1}^{i} s_l$
$OPT(i, 1) = \{s_1, ..., s_i\}$

Step Four: Iterative DP

Algorithm 1 Iterative DP to find all possible sums of subset

```
1:  procedure FindSums(d, f)
2:      n = |d|
3:      Calculate s_i = d_i − f_i
4:      Calculate total sum S = Σ_{i=1}^{n} s_i
5:      Initialize memo[j][k] = ∅ for j, k in {1, ..., n}
6:      Set memo[i][i] = Σ_{l=1}^{i} s_l
7:      Set memo[i][1] = {s_1, ..., s_i}
8:      for i = 2 to n  do
9:          for j = 1 to i do
10:             tempSet = ∅
11:             for e in memo[i-1][j] do
12:                 tempSet = tempSet ∪ {e + s_i}
13:             end for
14:             memo[i][j] = memo[i − 1][j] ∪ tempSet
15:         end for
16:     end for
17:     for each sum in memo[n][n/2] do
18:         if 0 < sum < S then
19:             return True
20:         end if
21:     end forreturn False
22:
23: end procedure
```

Correctness:

By induction on recurrence. If $OPT(i, j)$ correctly captures all possible sums using j elements from first $\{s_1, ..., s_i\}$, then from the recurrence, $OPT(i, j+1)$ denotes all possible sums taking one more $(j + 1)$ elements in subset $\{s_1, ..., s_i\}$, and $[OPT(i, j) + s_{i+1}]$ denotes all possible sums in $OPT(i, j)$ added by s_{i+1}. The two components ensures that all possible sums with $j + 1$ elements are added from $\{s_1, ..., s_{i+1}\}$, without and with s_{i+1}. Thus, the recurrence is correct.

Runtime Analysis:

Each iteration of inner two for loops takes $O(n)$. Assuming union operation takes constant time for quick-union, or $O(\log(n))$ for standard union as in union-find. Initialization of summation and memo table set up takes $O(n^2)$. Thus, the algorithm takes $O(n^2)$ in total using quick-union operations, or $O(n^2 \log(n))$ for standard union as in union-find.