Name: Ahmed Elsayed

Project 2 Report

1. **Does the branch-and-bound solution always perform better than the backtracking solution?**

Yes. According to my testing, brand and bound perform efficiently as less checked nodes wise, however; it takes more time to process than backtracking

1. **what input did the backtracking solution check less nodes?**

I added a table below showing that the smaller the total weight required becomes, the less nodes that must be checked.

1. **Did you find any inputs for which the brute-force algorithm is not much worse than either of the other two solutions?**

Brute force tends to process faster than branch and bound in my testing but not as efficient as the other two solutions

1. **On average, how much more efficient are the two solutions discussed in class over the brute-force solution?**

For some reason, brute force tends to act faster than branch and bound in my testing

1. **After running your program on several different inputs, make a table displaying the # of nodes checked for each of the three solutions.**

Input: p {40,30,50,10}, w {2,5,10,5} W = 16

|  |  |  |  |
| --- | --- | --- | --- |
|  | Backtracking | Branch and Bound | Brute Force |
| Time duration | 1036496 | 1933079 | 1396130 |
| No of checked nodes | 13 | 11 | 31 |

Input: p {102,20,60,40}, w {3,2,4,1}, W= 5

|  |  |  |  |
| --- | --- | --- | --- |
|  | Backtracking | Branch and Bound | Brute Force |
| Time duration | 1234905 | 2276215 | 1121047 |
| No of checked nodes | 9 | 4 | 31 |

Input: p {20,30,35,12,3}, w {2,5,7,3,1}, W= 13

|  |  |  |  |
| --- | --- | --- | --- |
|  | Backtracking | Branch and Bound | Brute Force |
| Time duration | 1314099 | 2113335 | 1013532 |
| No of checked nodes | 17 | 9 | 63 |