Homework

(Due Nov. 2, 2020)

(1) Given the following tableau, determine the optimal selection of items in a 0-1 knapsack problem. The weights and values are

	A	B	C	D	E	F	G	H
Weight	10	9	8	7	6	5	4	3
Value	12	10	9	8	8	6	5	4

Do this for three different maximum capacities, namely 20, 30, and 35.

Capacity	A	A,B	A,B,C	A,B,C,D	A, …, E	A, … , F	A, … , G	A, … ,H
0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	4
4	0	0	0	0	0	0	5	5
5	0	0	0	0	0	6	6	6
6	0	0	0	0	8	8	8	8
7	0	0	0	8	8	8	8	9
8	0	0	9	9	9	9	9	10
9	0	10	10	10	10	10	11	12
10	12	12	12	12	12	12	13	13
11	12	12	12	12	12	14	14	14
12	12	12	12	12	12	14	14	15
13	12	12	12	12	16	16	16	17
14	12	12	12	12	17	17	17	18
15	12	12	12	17	18	18	19	19
16	12	12	12	18	20	20	20	20

17	12	12	19	20	20	20	21	21
18	12	12	21	21	21	22	22	23
19	12	22	22	22	22	23	23	24
20	12	22	22	22	22	24	25	25
21	12	22	22	22	25	26	26	26
22	12	22	22	22	26	26	27	27
23	12	22	22	22	28	28	28	29
24	12	22	22	27	29	29	29	30
25	12	22	22	29	30	30	31	31
26	12	22	22	30	30	31	31	32
27	12	22	31	31	31	32	33	33
28	12	22	31	31	31	34	34	35
29	12	22	31	31	31	35	35	35
30	12	22	31	31	35	36	36	37
31	12	22	31	31	37	37	37	38
32	12	22	31	31	38	38	39	39
33	12	22	31	31	39	39	40	40
34	12	22	31	39	39	39	41	41
35	12	22	31	39	39	41	42	43

(2) Develop a dynamic programming solution to finding the sum of the largest contiguous subarray in an array of integers $[a_1, a_2, a_3, \ldots, a_n]$ that has the maximum sum. The solution should be linear in n and therefore beat the divide-and-conquer solution we have looked at in session of $O(n \log(n))$. Explain why your solution identifies the sum.