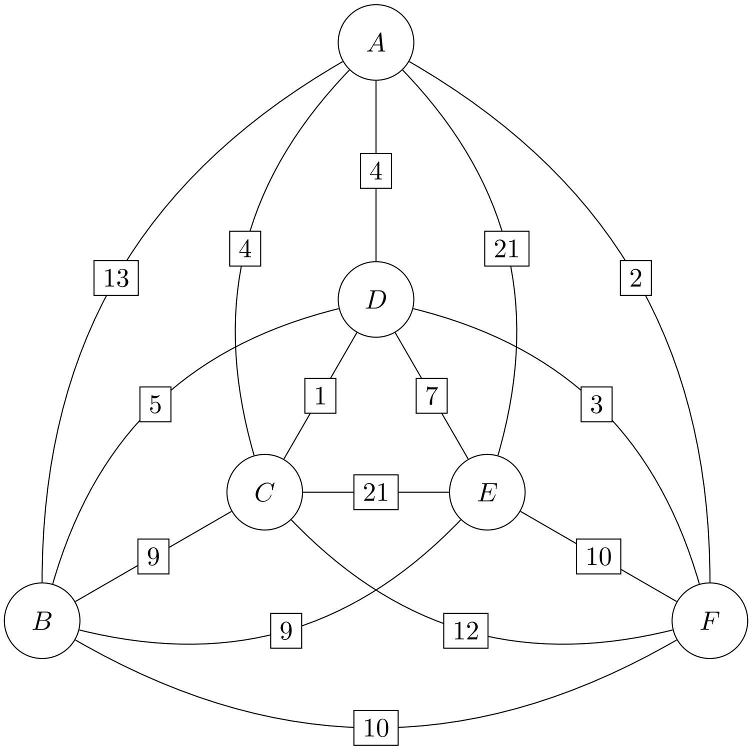
5CCS2FC2: Foundations of Computing II

Tutorial Sheet 9

9.1 Consider the following complete weighted graph:



Apply the *2-opt optimisation algorithm* to find a local optimum traversal.

9.2 Consider the Greedy Knapsack algorithm from week 6:

```
Algorithm Greedy Knapsack
  Sort item_list by $value/weight$ ratio
  for all item in item_list do
    if current_capacity + $weight$(item) $<$ max_capacity then
      add item to knapsack
    end if
  end for
```

Apply the Greedy Knapsack algorithm to each of the following Knapsack instances:

(i) Number of items: 4 Knapsack size: 12

item_list	A	B	C	D	Total
weight	7	1	1	11	20
value	3	1	3	11	18

(ii) Number of items: 4 Knapsack size:6

item_list	A	B	C	D	Total
weight	2	3	5	1	11
value	17	4	20	11	52

(iii) Number of items: 5 Knapsack size: 14

item_list	A	B	C	D	E	Total
weight	9	1	1	4	13	28
value	2	2	2	2	18	26

Find the *optimal* knapsack selection (by any method) and calculate the *approximation ratio* for each of the above instances.

9.3 *(tricky!)* What is the *worst case* approximation ratio for this implementation of the Greedy Knapsack algorithm?