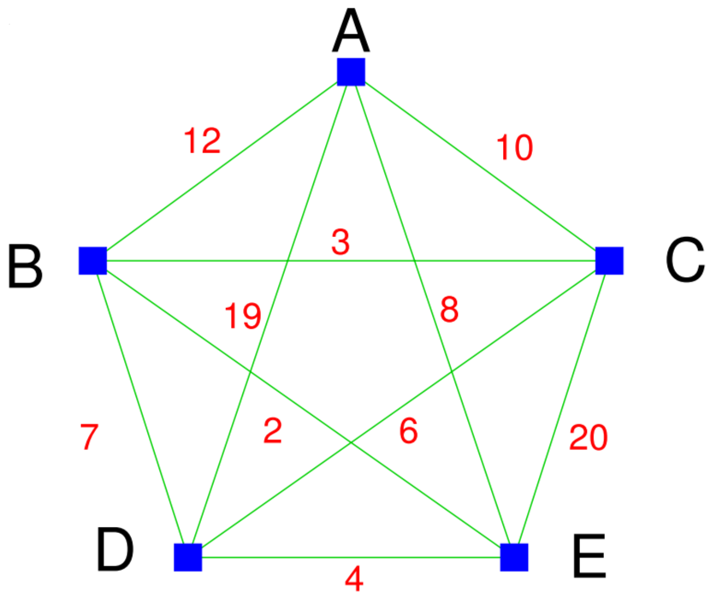
Es. 1

1. Apply the Nearest Neighbor (NN) and Savings (S) heuristic, using A as starting node.
2. Apply a steepest descent local search using as starting solution the one obtained with S. Use the neighborhood defined as «swap of adjacent cities in the solution».



$$C = \begin{bmatrix} 0 & 12 & 10 & 19 & 8 \\ 12 & 0 & 3 & 7 & 2 \\ 10 & 3 & 0 & 6 & 20 \\ 19 & 7 & 6 & 0 & 4 \\ 8 & 2 & 20 & 4 & 0 \end{bmatrix}$$

Es. 2

Consider this starting solution for a symmetric TSP:

S = 1-2-3-4-(1).

Apply a “First Improvement” local search, until the local minima. Repeat the exercise two times using as neighborhoods:

1) Swap adjacent nodes.
2) 2-opt.

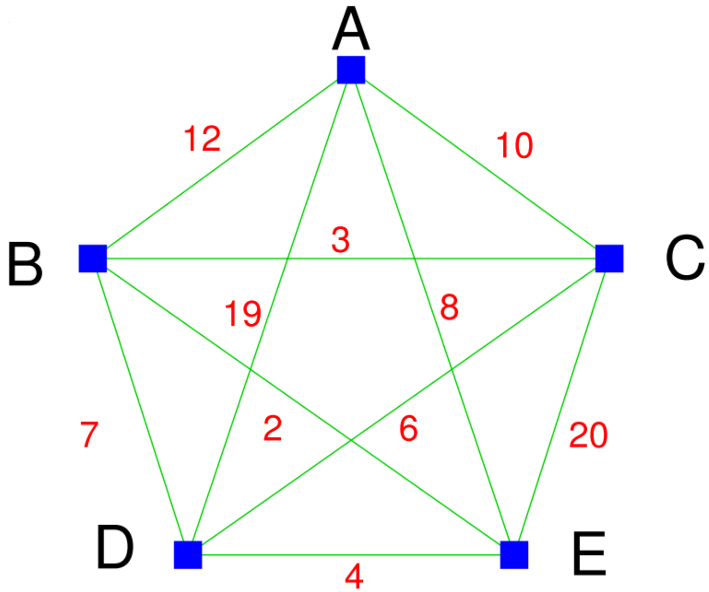
	1	**2**	**3**	**4**
1	0	4	2	4
2	4	0	4	3
3	2	4	0	3
4	4	3	3	0

Es. 3

1) Apply Nearest Neighbor and Savings.
2) Starting from the best sequence obtained among the two, apply a steepest descent local search, using a «swap adjacent cities» neighborhood.

	1	2	3	4	5
1	0	3	7	1	5
2	3	0	2	5	4
3	7	2	0	8	3
4	1	5	8	0	6
5	5	4	3	6	0

Solutions

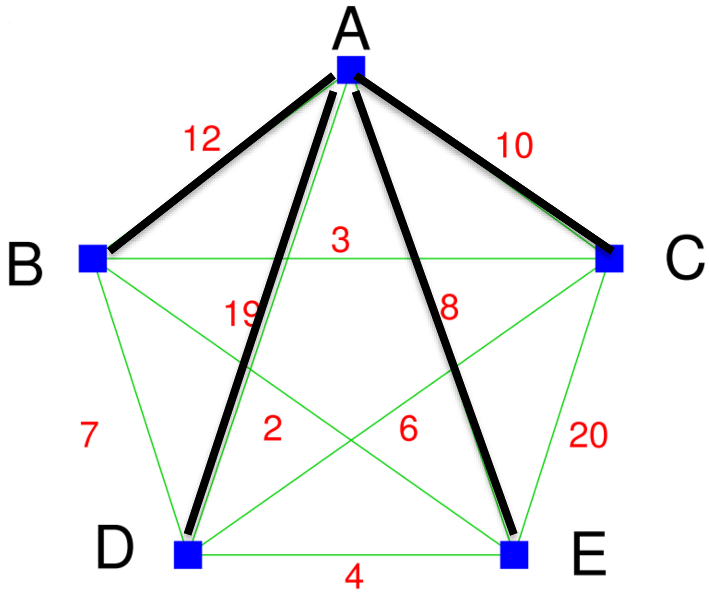


$$C = \begin{bmatrix} 0 & 12 & 10 & 19 & 8 \\ 12 & 0 & 3 & 7 & 2 \\ 10 & 3 & 0 & 6 & 20 \\ 19 & 7 & 6 & 0 & 4 \\ 8 & 2 & 20 & 4 & 0 \end{bmatrix}$$

Initial Node: A

Current Node	Nodes to be visited	Selected Arc
A	B,C,D,E	(A,E) with cost 8
E	B,C,D	(E,B) with cost 2
B	C,D	(B,C) with cost 3
C	D	(C,B) with cost 6
To close the cycle:		(D,A) with cost 19

Total Cost: 38



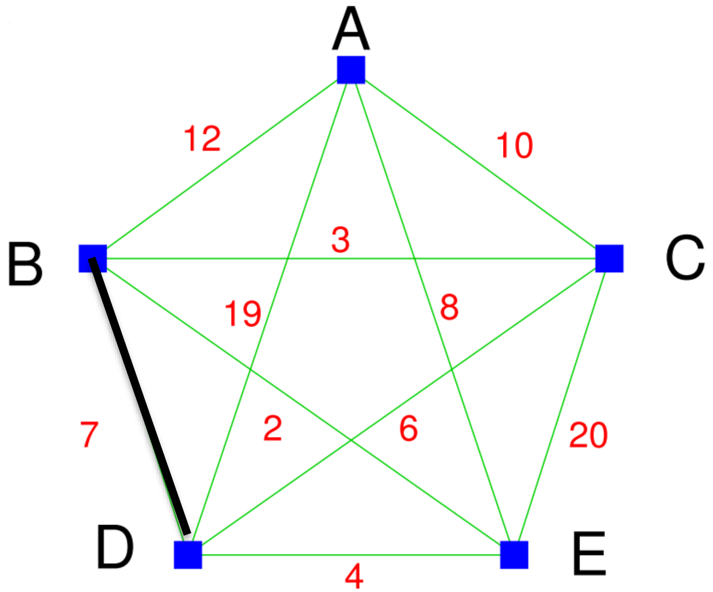
$$C = \begin{bmatrix} 0 & 12 & 10 & 19 & 8 \\ 12 & 0 & 3 & 7 & 2 \\ 10 & 3 & 0 & 6 & 20 \\ 19 & 7 & 6 & 0 & 4 \\ 8 & 2 & 20 & 4 & 0 \end{bmatrix}$$

Initial Node = A

s(i,j) = c(k,i) + c(j,k) - c(i,j)

$$s_{BC} = c_{AB} + c_{AC} - c_{BC} = 12 + 10 - 3 = 19$$
$$s_{BD} = c_{AB} + c_{AD} - c_{BD} = 12 + 19 - 7 = 24$$
$$s_{BE} = c_{AB} + c_{AE} - c_{BE} = 12 + 8 - 2 = 18$$
$$s_{CD} = c_{AC} + c_{AD} - c_{CD} = 10 + 19 - 6 = 23$$
$$s_{CE} = c_{AC} + c_{AE} - c_{CE} = 10 + 8 - 20 = -2$$
$$s_{DE} = c_{AD} + c_{AE} - c_{DE} = 19 + 8 - 4 = 23$$

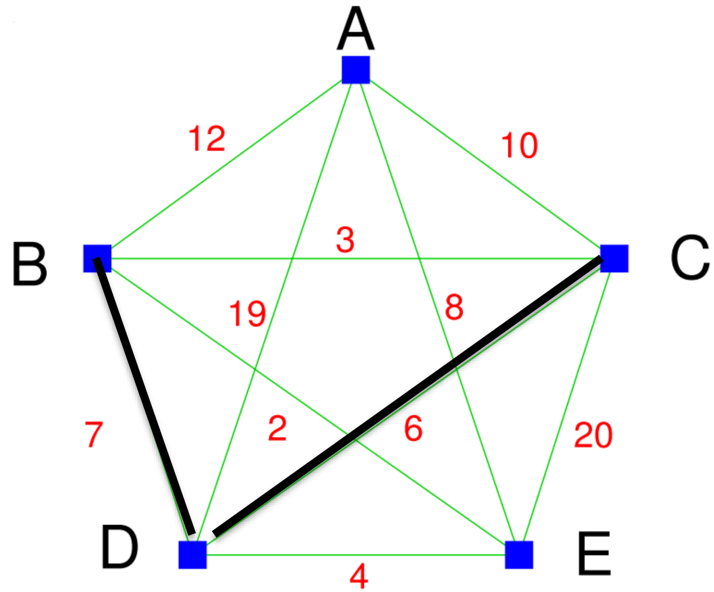
Ordered:
BD
CD
DE
BC
BE
CE



$$C = \begin{bmatrix} 0 & 12 & 10 & 19 & 8 \\ 12 & 0 & 3 & 7 & 2 \\ 10 & 3 & 0 & 6 & 20 \\ 19 & 7 & 6 & 0 & 4 \\ 8 & 2 & 20 & 4 & 0 \end{bmatrix}$$

Initial Node = A

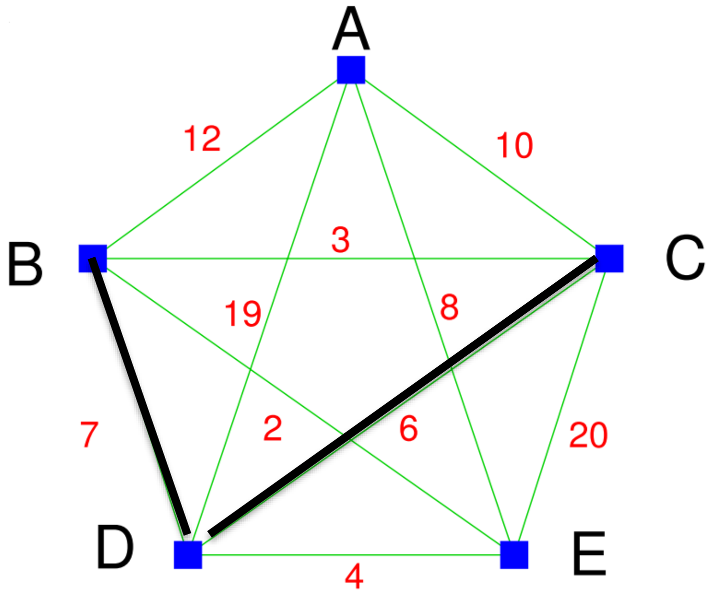
BD : Can be added
CD
DE
BC
BE
CE



$$C = \begin{bmatrix} 0 & 12 & 10 & 19 & 8 \\ 12 & 0 & 3 & 7 & 2 \\ 10 & 3 & 0 & 6 & 20 \\ 19 & 7 & 6 & 0 & 4 \\ 8 & 2 & 20 & 4 & 0 \end{bmatrix}$$

Initial Node = A

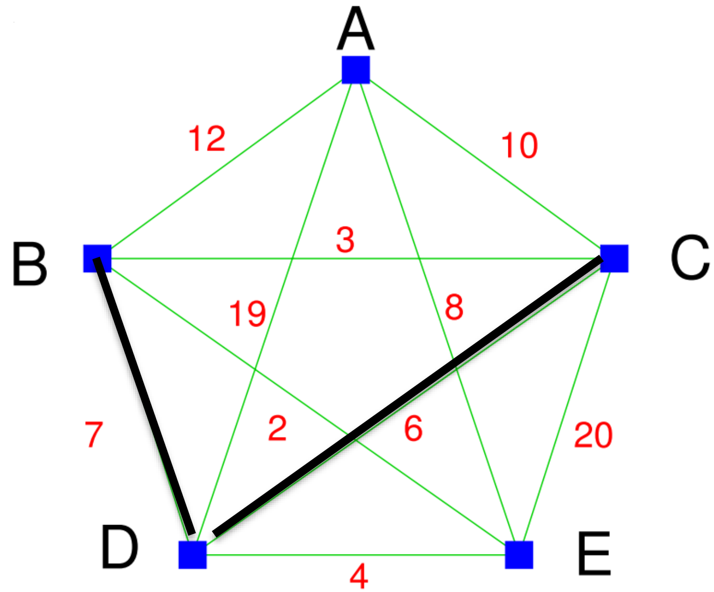
BD : Can be added
CD : Can be added
DE
BC
BE
CE



$$C = \begin{bmatrix} 0 & 12 & 10 & 19 & 8 \\ 12 & 0 & 3 & 7 & 2 \\ 10 & 3 & 0 & 6 & 20 \\ 19 & 7 & 6 & 0 & 4 \\ 8 & 2 & 20 & 4 & 0 \end{bmatrix}$$

Initial Node = A

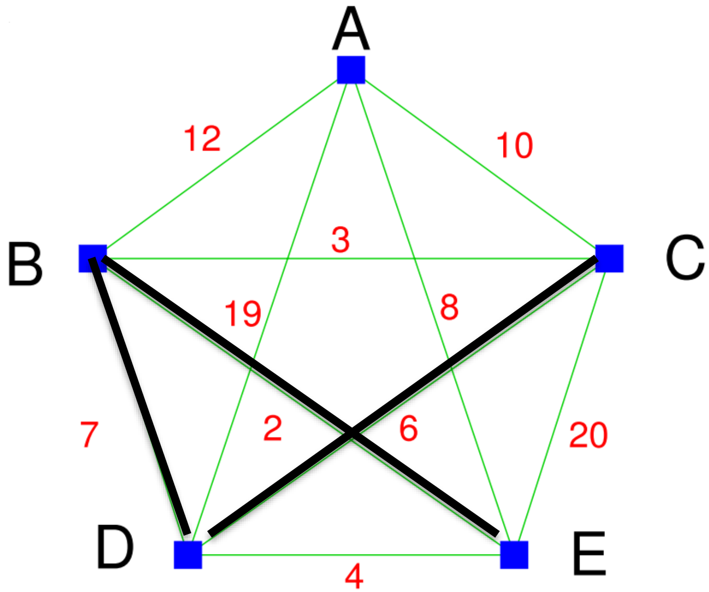
BD : Can be added
CD : Can be added
DE : NO (D already have input and output arc)
BC
BE
CE



$$C = \begin{bmatrix} 0 & 12 & 10 & 19 & 8 \\ 12 & 0 & 3 & 7 & 2 \\ 10 & 3 & 0 & 6 & 20 \\ 19 & 7 & 6 & 0 & 4 \\ 8 & 2 & 20 & 4 & 0 \end{bmatrix}$$

Initial Node = A

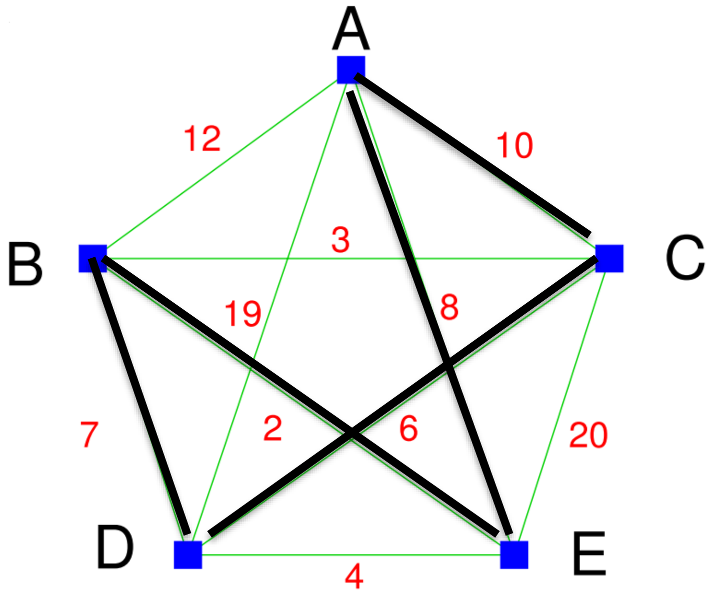
BD : Can be added
CD : Can be added
DE : NO
BC : NO (it would create a cycle without all cities)
BE
CE



$$C = \begin{bmatrix} 0 & 12 & 10 & 19 & 8 \\ 12 & 0 & 3 & 7 & 2 \\ 10 & 3 & 0 & 6 & 20 \\ 19 & 7 & 6 & 0 & 4 \\ 8 & 2 & 20 & 4 & 0 \end{bmatrix}$$

Initial Node = A

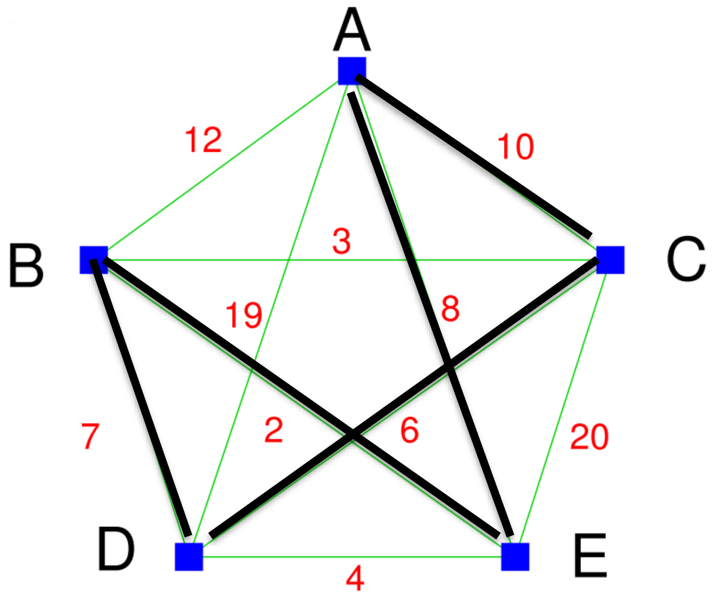
BD : Can be added
CD : Can be added
DE : NO
BC : NO
BE : Can be added
CE :



$$C = \begin{bmatrix} 0 & 12 & 10 & 19 & 8 \\ 12 & 0 & 3 & 7 & 2 \\ 10 & 3 & 0 & 6 & 20 \\ 19 & 7 & 6 & 0 & 4 \\ 8 & 2 & 20 & 4 & 0 \end{bmatrix}$$

Objective Function = 10+6+7+2+8 = 33

Local search

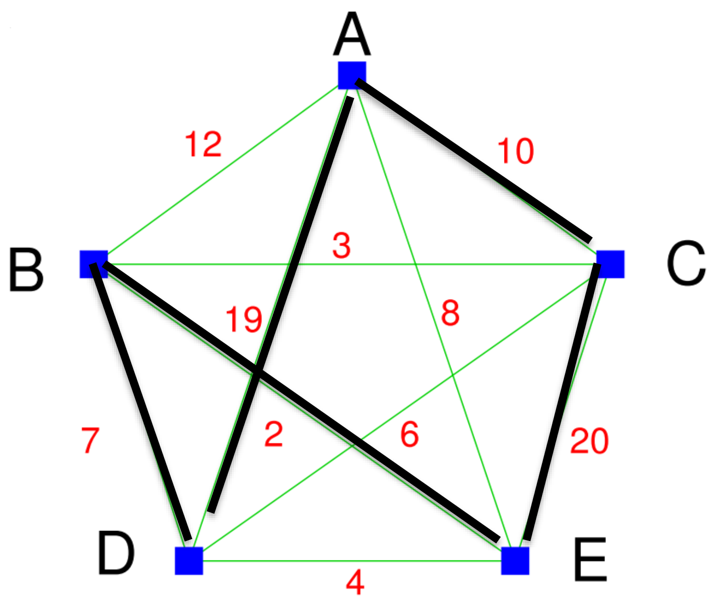


Starting sequence: ACDBE (cost 33)

Neighbors:

CADBE
A**DC**BE
AC**BD**E
ACD**EB**
ECDB**A**

Local search



Starting sequence: ACDBE (cost 33)

Neighbors:

CADBE : cost 58

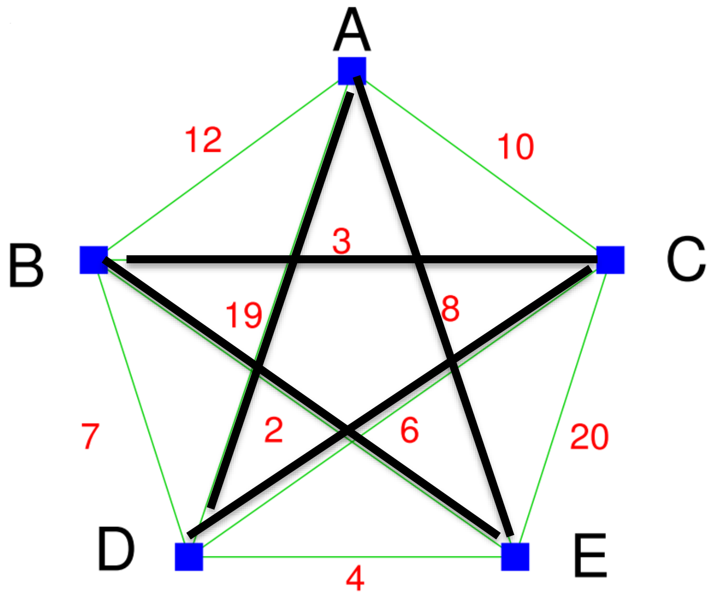
A**DC**BE

AC**BD**E

ACD**EB**

ECDB**A**

Local search

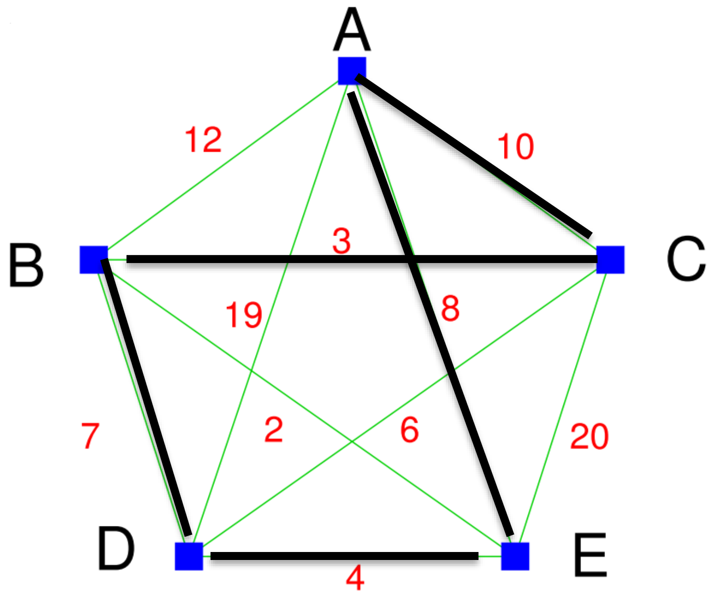


Starting sequence: ACDBE (cost 33)

Neighbors:

CADBE : cost 58
A**DC**BE : cost 38
AC**BD**E
ACD**EB**
ECDB**A**

Local search

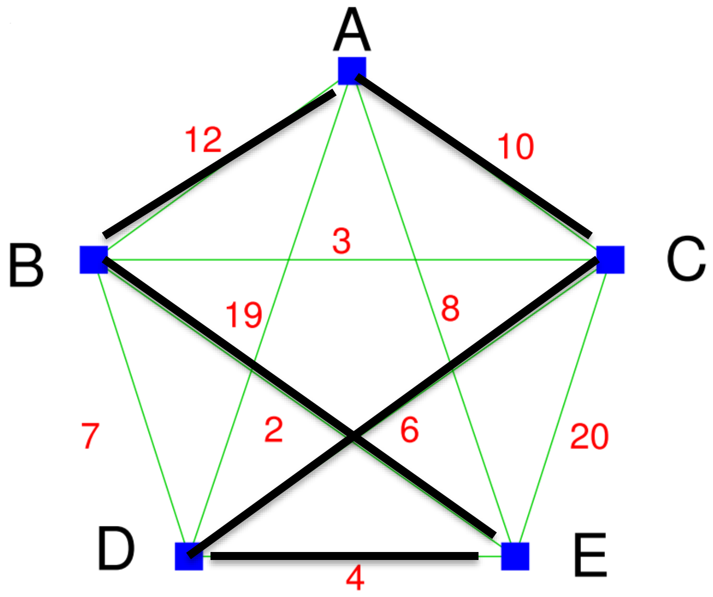


Starting sequence: ACDBE (cost 33)

Neighbors:

CADBE : cost 58
A**DC**BE : cost 38
AC**BD**E : cost 32 (improving!!!!)
ACD**EB**
ECDB**A**

Local search

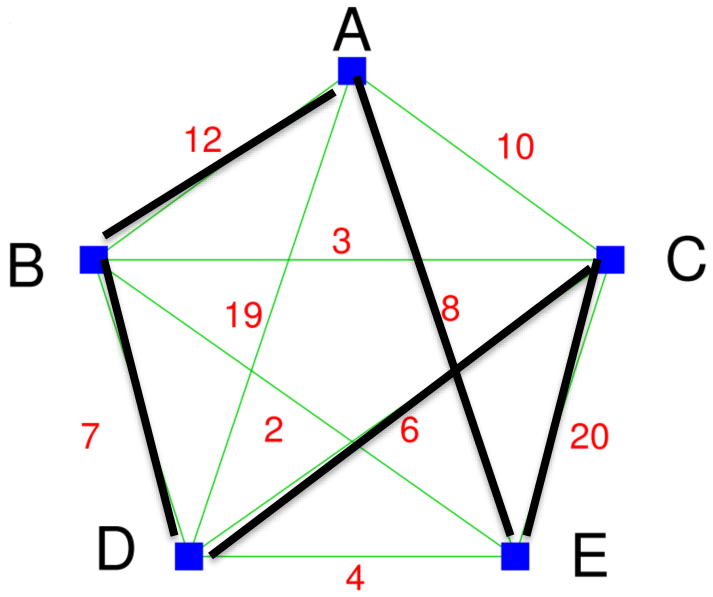


Starting sequence: ACDBE (cost 33)

Neighbors:

CADBE : cost 58
A**DC**BE : cost 38
AC**BD**E : cost 32
ACD**EB** : cost 34
ECDB**A**

Local search



Sequenza originale: ACDBE (costo 33)

Vicini:

CADBE : cost 58
A**DC**BE : cost 38
AC**BD**E : cost 32
ACD**EB** : cost 34
ECDB**A** : cost 53

Starting sequence: ACDBE (cost 33)

The best neighbor is

ACBDE with objective function = 32

This solution become the new current solution.

A new iteration (left as exercise) will not produce any improvement.

Solution 2

$F(1,2,3,4) = 15$ = Current sol

$F(\boldsymbol{2,1},3,4) =$
$F(1,\boldsymbol{3,2},4) =$
$F(1,2,\boldsymbol{4,3}) =$
$F(\boldsymbol{4},2,3,\boldsymbol{1}) =$

	1	**2**	**3**	**4**
1	0	4	2	4
2	4	0	4	3
3	2	4	0	3
4	4	3	3	0

$F(1,2,3,4) = 15$ = Current sol.

$F(\mathbf{2,1},3,4) =$ 12 (improving!)
$F(1,\mathbf{3,2},4) =$
$F(1,2,\mathbf{4,3}) =$
$F(\mathbf{4},2,3,\mathbf{1}) =$

	1	**2**	**3**	**4**
1	0	4	2	4
2	4	0	4	3
3	2	4	0	3
4	4	3	3	0

The first solution of the neighborhood improves the current one. In a "First Improvement" local search, it becomes the new current solution (we do not explore the remaining neighbors)

$F(2,1,3,4) = 12$ = Current sol.

$F(\boldsymbol{1,2},3,4) = 15$
$F(2,\boldsymbol{3,1},4) = 13$
$F(2,1,\boldsymbol{4,3}) = 15$
$F(\boldsymbol{4},1,3,\boldsymbol{2}) = 13$

	1	**2**	**3**	**4**
1	0	4	2	4
2	4	0	4	3
3	2	4	0	3
4	4	3	3	0

No improving solutions.
The algorithm stops.

F(*1*,*2*,*3*,*4*) = 15 = Current sol.

2-opt neighbors are n(n-3)/2*
With n=4, we have 2 neighbors only:

(1,3) F(***3,2***,***1***,*4*) = 15
(2,4) F(*1*,***4,3***,***2***) = 15

No improvements, the local search stops.

	1	**2**	**3**	**4**
1	0	4	2	4
2	4	0	4	3
3	2	4	0	3
4	4	3	3	0

Solution 3

1) NN solution = 1-4-2-3-5-(1) with cost 16
2) SAVINGS solution = 1-2-3-5-4-(1) with cost 15
3) No improving solutions already at the first iteration. The local minima solution is 1-2-3-5-4-(1) with cost 15

	1	2	3	4	5
1	0	3	7	1	5
2	3	0	2	5	4
3	7	2	0	8	3
4	1	5	8	0	6
5	5	4	3	6	0