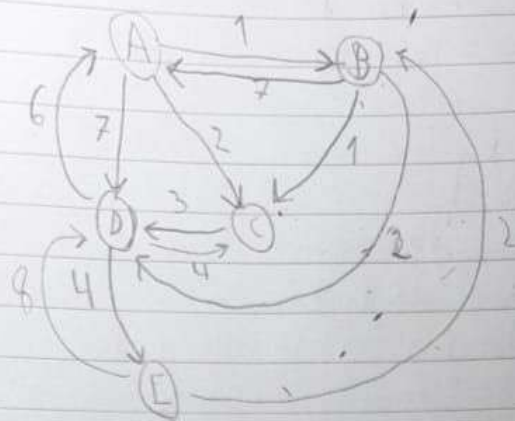


PD 7, ej 1:



S = Nodos y
Visitados

S	Actual	D[A]	D[B]	D[C]	D[D]	D[E]
{A}	B	7	0	1	2	∞
{B, C}	C	7	0	1	2	∞
{B, C, D}	D	7	0	1	2	6
{B, C, D, E}	E	7	0	1	2	6
{A, C, D, E}	A	7	0	1	2	6

Algoritmo:

COM:

B, C, D, E, A - A

Construir Camino (Vertices, inicio, destino)

lista camino ← inicio

Si: inicio == destino

imprimir destino

Fin Si

Verice Final ← destino

Ejercicio 2:

A \rightarrow C (1) \rightarrow D (4)

B \rightarrow E (3) \rightarrow A (6)

C \rightarrow E (1) \rightarrow B (2)

D \rightarrow C (5)

E \rightarrow A (3)

b) Menores costos.

	A	B	C	D	E
A	0	3	1	4	2
B	6	0	7	10	3
C	8	2	0	8	1
D	9	7	5	0	6
E	3	6	4	7	0

Excentricidad:

A: 4 \rightarrow Centro del grafo

B: 10

C: 8

D: 9

E: 7

Algoritmo:

Construir Camino (predecesores, inicio, destino)

lista camino \leftarrow vacía

Si inicio == destino

imprimir destino

Devolver a destino

nodo \leftarrow destino

Mientras nodo != origen hacer

lista camino.add(nodo)

Si predecesores[nodo] = nulo Entonces

Escribir "No hay camino encontrado"

Retornar

Fin Si

nodo \leftarrow predecesores[nodo]

Fin Mientras

lista camino.add(origen)

Invertir (lista camino)

Devolver listacaminos

Ejercicio 3:

g) $A: A \rightarrow B(4) \rightarrow C(3) \rightarrow D(1)$

$B: B \rightarrow F(4)$

$C: C \rightarrow E(7)$

$D: D \rightarrow E(5)$

$E: E \rightarrow F(3)$

$F: F \rightarrow C(2)$

h) $\{ A \rightarrow F, A \rightarrow E, B \rightarrow C, D \rightarrow F, C \rightarrow F, F \rightarrow E, E \rightarrow C \}$