Fig. 1

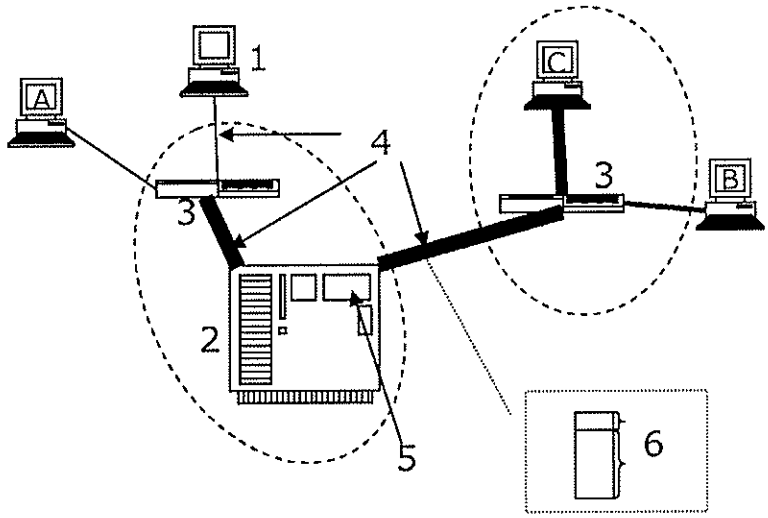


Fig. 2

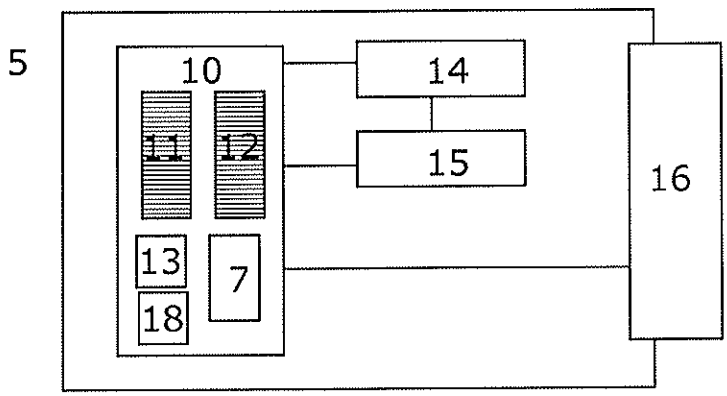


Fig. 3

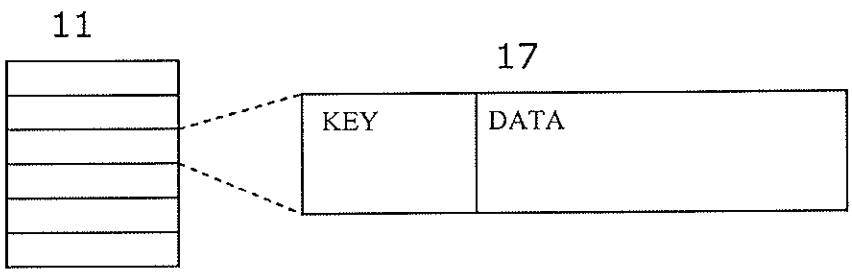


Fig. 4

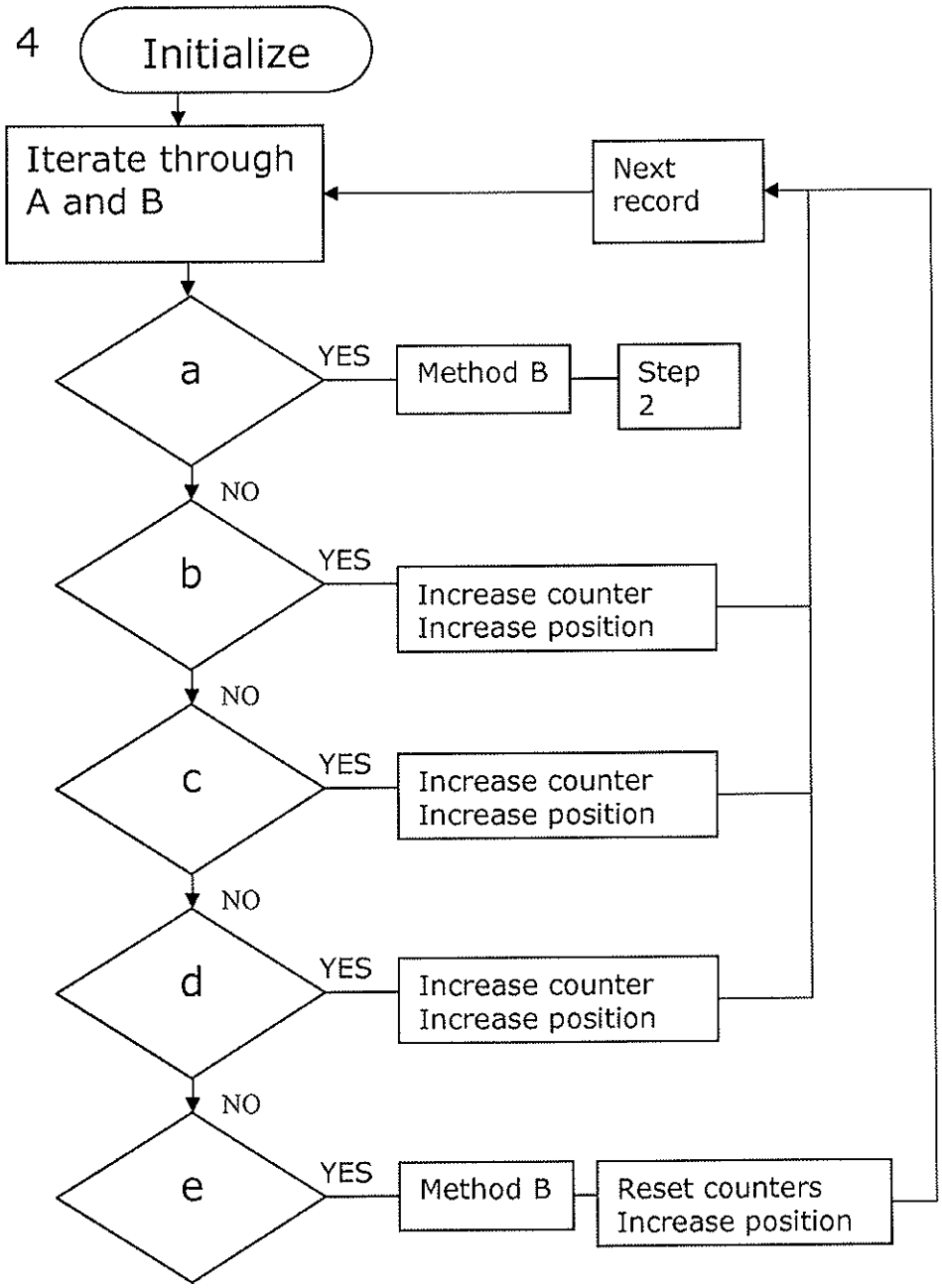

Initialize
Iterate through A and B
Next record
a
YES
Method B
Step 2
NO
b
YES
Increase counter
Increase position
NO
c
YES
Increase counter
Increase position
NO
d
YES
Increase counter
Increase position
NO
e
YES
Method B
Reset counters
Increase position

FIG 5

Data Set A

Logical Position	Key	Data part
1	99	10
2	98	10
3	97	10
4	96	10
5	95	10
6	94	10
7	93	10
8	92	10

Data Set B

Logical Position	Key	Data part
1	101	10
2	100	10
3	97	10
4	93	15

FIG 6

Data Set A

Logical Position	Key	Data part
1	99	10

Data Set B

Logical Position	Key	Data part
1	101	10

FIG 7

Data Set A

Logical Position	Key	Data part
1	99	10

Data Set B

Logical Position	Key	Data part
2	100	10

FIG 8

Data Set A

Logical Position	Key	Data part
1	99	10

Data Set B

Logical Position	Key	Data part
3	97	10

FIG 9

Data Set A

Logical Position	Key	Data part
2	98	10

Data Set B

Logical Position	Key	Data part
3	97	10

FIG 10

Data Set A

Logical Position	Key	Data part
3	97	10

Data Set B

Logical Position	Key	Data part
3	97	10

FIG 11

Operator	Logical Position	Key	Data part
Replace	1	101	10
Replace	2	100	10

FIG 12

Data Set A

Logical Position	Key	Data part
4	96	10

Data Set B

Logical Position	Key	Data part
4	93	15

FIG 13

Data Set A

Logical Position	Key	Data part
5	95	10

Data Set B

Logical Position	Key	Data part
4	93	15

FIG 14

Data Set A

Logical Position	Key	Data part
6	94	10

Data Set B

Logical Position	Key	Data part
4	93	15

FIG 15

Data Set A

Logical Position	Key	Data part
7	93	10

Data Set B

Logical Position	Key	Data part
4	93	15

FIG 16

Data Set A

Logical Position	Key	Data part
8	92	10

Data Set B

Logical Position	Key	Data part
5		

FIG 17

Data Set A

Logical Position	Key	Data part
9		

Data Set B

Logical Position	Key	Data part
5		

FIG 18

Operator	Logical Position	Key	Data part
Delete (5)	4		
Add	4	93	15

FIG 19

Operator	Logical Position	Key	Data part
Replace	1	101	10
Replace	2	100	10
Delete (5)	4		
Add	4	93	15

FIG 20

Data Set A

Logical Position	Key	Data part
1	99	10
2	98	10
3	97	10
4	96	10
5	95	10
6	94	10
7	93	10
8	92	10

Data Set B

Logical Position	Key	Data part
1	101	10
2	100	10
3	97	10
4	93	15

FIG 21

Operator	Logical Position	Key	Data part
Replace	1	101	10

FIG 22

Data Set A

Logical Position	Key	Data part
1	~~99~~	~~10~~
	101	**10**
2	98	10
3	97	10
4	96	10
5	95	10
6	94	10
7	93	10
8	92	10

FIG 23

Operator	**Logical Position**	**Key**	**Data part**
Replace	2	100	10

FIG 24

Data Set A

Logical Position	**Key**	**Data part**
1	~~99~~	~~10~~
	101	10
2	~~98~~	~~10~~
	100	**10**
3	97	10
4	96	10
5	95	10
6	94	10
7	93	10
8	92	10

FIG 25

Operator	**Logical Position**	**Key**	**Data part**
Delete (5)	4		

FIG 26

Data Set A

Logical Position	**Key**	**Data part**
1	~~99~~	~~10~~
	101	10
2	~~98~~	~~10~~
	100	10
3	97	10
~~4~~	**~~96~~**	**~~10~~**
~~5~~	**~~95~~**	**~~10~~**
~~6~~	**~~94~~**	**~~10~~**
~~7~~	**~~93~~**	**~~10~~**
~~8~~	**~~92~~**	**~~10~~**

FIG 27

Operator	**Logical Position**	**Key**	**Data part**
Add	4	93	15

FIG 28

Data Set A

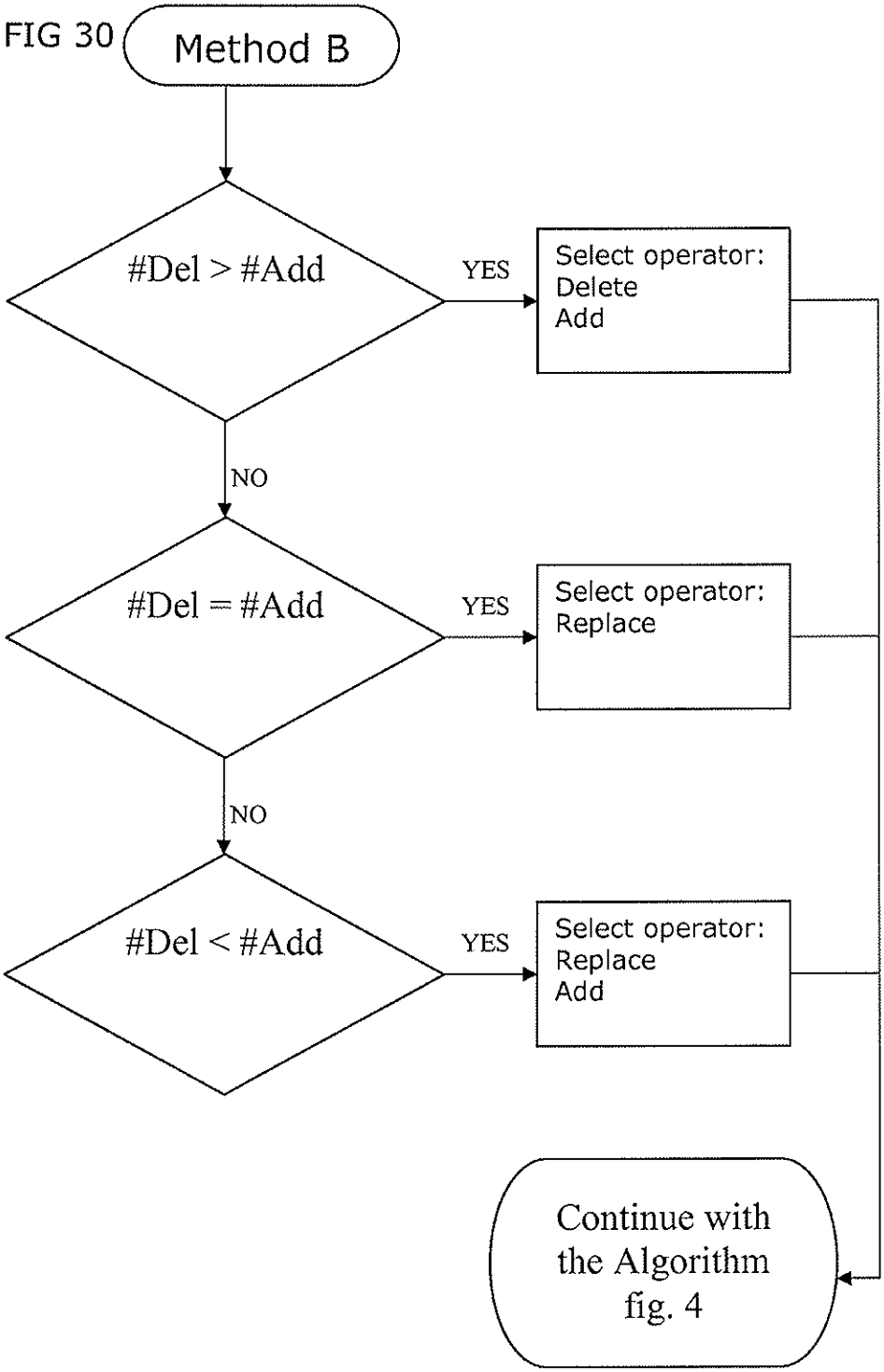
Logical Position	Key	Data part
1	~~99~~	~~10~~
	101	10
2	~~98~~	~~10~~
	100	10
3	97	10
4	~~96~~	~~10~~
	93	**15**
5	~~95~~	~~10~~
6	~~94~~	~~10~~
7	~~93~~	~~10~~
8	~~92~~	~~10~~

FIG 29

Data Set B

Logical Position	Key	Data part
1	101	10
2	100	10
3	97	10
4	93	15

FIG 30


Method B
#Del > #Add
YES
Select operator:
Delete
Add
NO
#Del = #Add
YES
Select operator:
Replace
NO
#Del < #Add
YES
Select operator:
Replace
Add
Continue with
the Algorithm
fig. 4