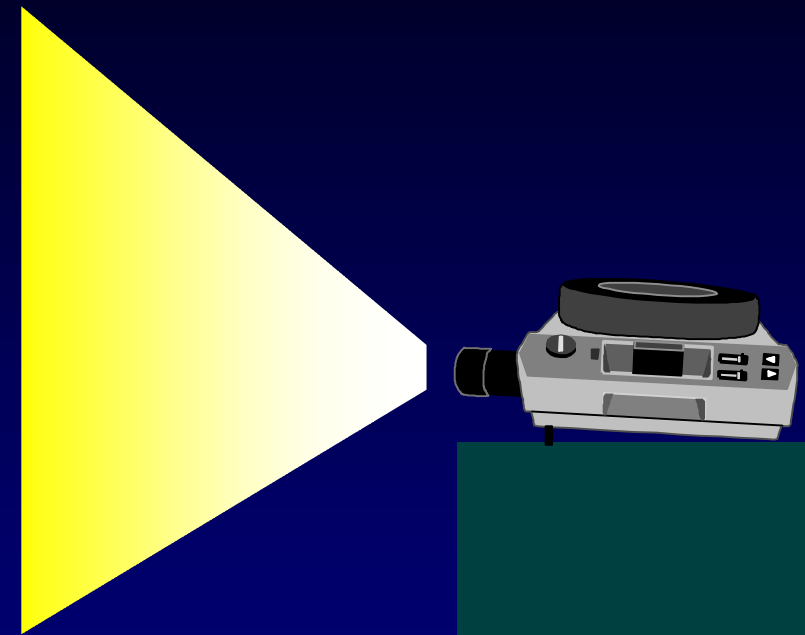


# Sequential Files : Outline

- Overview
- Ordered vs. Unordered
- Physical sequential Files
- Physical Linked Sequential Files
- COBOL



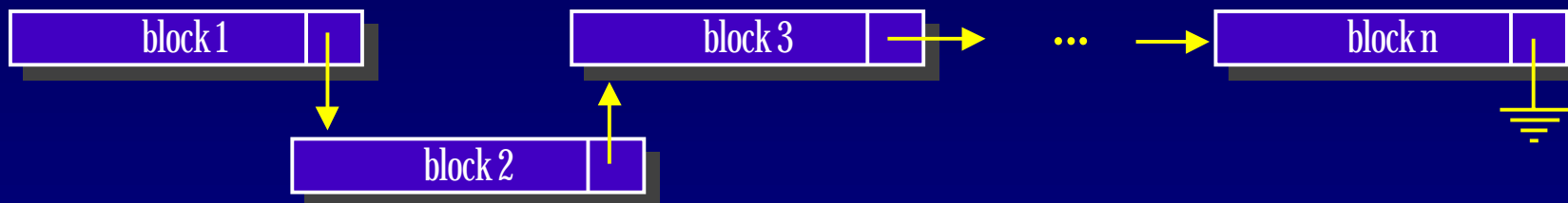
# Sequential Files

▼ Logically the records are stored consecutively

– Physical sequential file



– Physical linked sequential file



*Batch and RetrieveAll*

# Ordered and Unordered

## ▼ Ordered sequential files

- the logical records are stored sequentially in increasing (or decreasing) order according to their primary key values.

## ▼ Unordered sequential files

- the logical records are stored no in any specific order.

# Notations

▼ Let  $S$  be a file

- $NR$  : the number of logical records
- $BF$  : the blocking factor
- $NBLK$  : the number of blocks

▼  $SL[operation, S]$  : search length

▼  $sba$  : sequential block access

▼  $rba$  : random block access

# Physical Sequential Files

## ▼ Tape files

- Unordered files
- Ordered files

## ▼ Disk files

- Unordered files
- Ordered files

# Tape Files : Unordered

$$SL[RetrieveAll, S] = NBLK \quad sba$$

$$SL_{avg}[RetrieveOne, S] = \frac{NBLK + 1}{2} \quad sba \quad (\text{found})$$

$$SL[RetrieveOne, S] = NBLK \quad sba \quad (\text{not found})$$

Never would one use unordered sequential file if one permits *InsertOne*, *DeleteOne*, and *UpdateOne* operations.

# Tape Files : Ordered

- ▼ Require initial sorting operation of the given file.
- ▼ Batch mode update processing :
  - *InsertOne, DeleteOne, and UpdateOne*
- ▼ Batch mode retrieve processing :
  - *Batch and RetrieveFew*

$$SL[\text{RetrieveAll}, S] = NBLK \quad sba$$
$$SL_{avg}[\text{RetrieveOne}, S] = \frac{NBLK + 1}{2} \quad sba \quad (\text{found})$$
$$SL[\text{RetrieveOne}, S] = NBLK \quad sba \quad (\text{not found})$$

# Transaction Files

▼ A sequential file contains all logical records that are to be either updated, deleted, and/or inserted into the master file.

– updated



– deleted

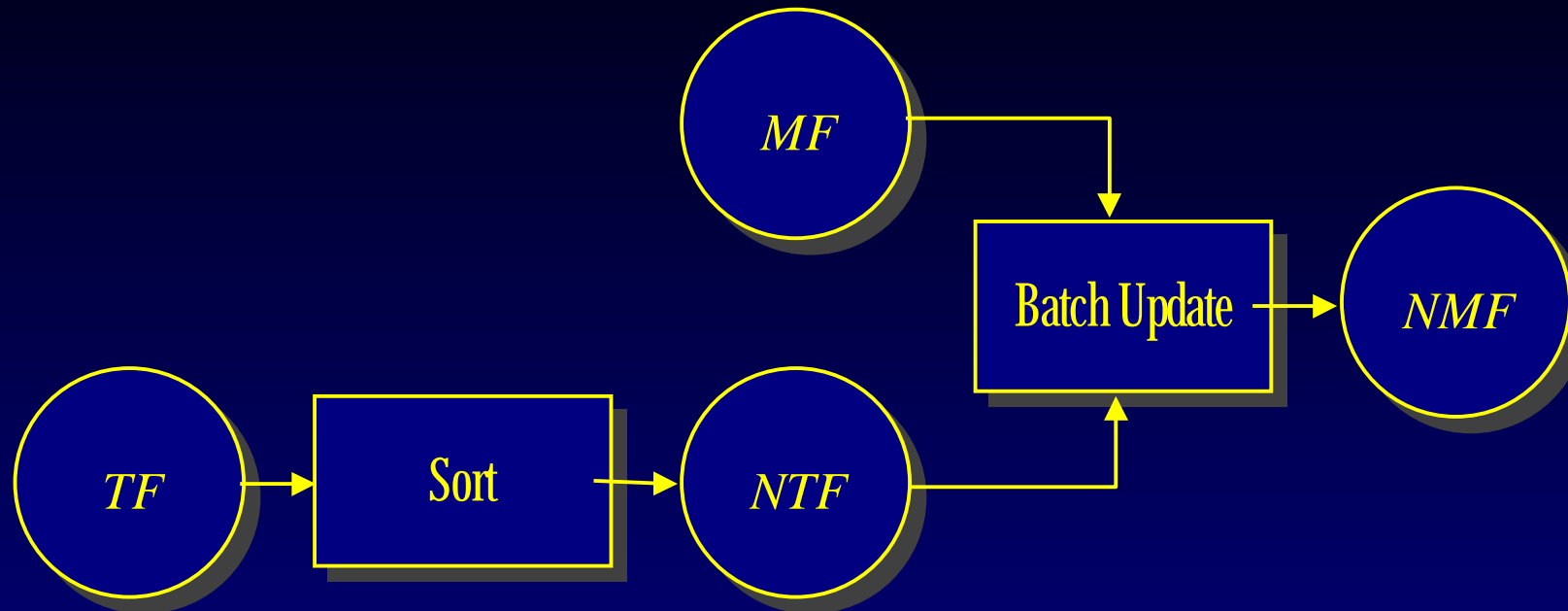


– inserted





# Batch Mode



- TF* : Transaction file
- NTF* : New transaction file (ordered)
- MF* : Master file (ordered)
- NMF* : New mater file (ordered)

# Batch Mode

D	125321	Somboon
I	233218	Somkuan
I	235089	Somhatai
U	255989	Somsak
I	817222	Somwai

*TF*

012834	Somchai
125321	Somboon
232117	Somkid
255989	Somsuk
592211	Somsri
877652	Somwang

*MF*

Batch Update

*NMF*

012834	Somchai
232117	Somkid
233218	Somkuan
235089	Somhatai
255989	Somsak
592211	Somsri
817222	Somwai
877652	Somwang

# Disk Files : Unordered

- ▼ Search length : same as the unordered tape files.
  - *RetrieveAll*
  - *RetrieveOne* (found)
  - *RetrieveOne* (not found)
- ▼ *UpdateOne* can be done in place.

$$SL_{avg}[UpdateOne, S] = 1 + \frac{1 + NBLK}{2} sba$$

# Unordered Disk Files : Delete

- ▼ *DeleteOne* : flagging the record
  - same search length as *UpdateOne*



$$SL_{avg}[UpdateOne, S] = 1 + \frac{1 + NBLK}{2} sba$$

# Unordered Disk Files : Insert

## ▼ *InsertOne*

- append : retrieve the last block
  - ▼ if there is space, rewrite the updated block
$$1 rba + 1 sba$$
  - ▼ if full, get next block and rewrite the block
$$1 rba + 2 sba$$
- allocate empty slot in each block
  - ▼ operation cost is the same as *DeleteOne*

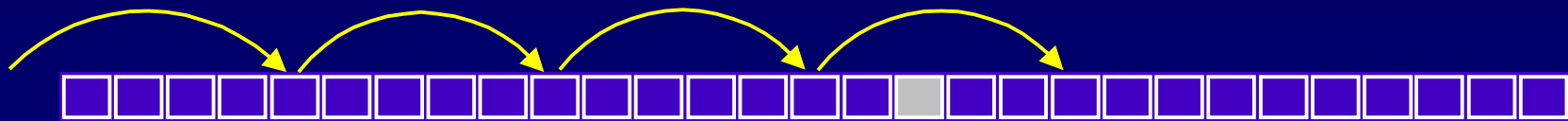


# Disk Files : Ordered

▼ *RetrieveAll* : *NBLK sba*

▼ *RetrieveOne* :

- sequential search
- binary search
- step search



▼ *UpdateOne, DeleteOne, InsertOne*\* :

- $1 sba + \text{the avg } SL \text{ of } RetrieveOne$

# Ordered Disk Files : *RetrieveOne*

## Sequential search

$$SL_{avg} [UpdateOne, S] = 1 + \frac{1 + NBLK}{M} \frac{1}{2} sba$$

## Binary search

$$SL_{avg} [UpdateOne, S] = 1 + \frac{1 + NBLK}{M} \frac{1}{2} sba$$

## Step search

$$SL_{avg} [UpdateOne, S] = 1 + \frac{1 + NBLK}{M} \frac{1}{2} sba$$

# Reorganization

- ▼ Deleting records by flagging
  - does not physically delete the records.
- ▼ Reorganization from time to time
  - Copying all unflagged records to a new file.
  - Purging the old file.



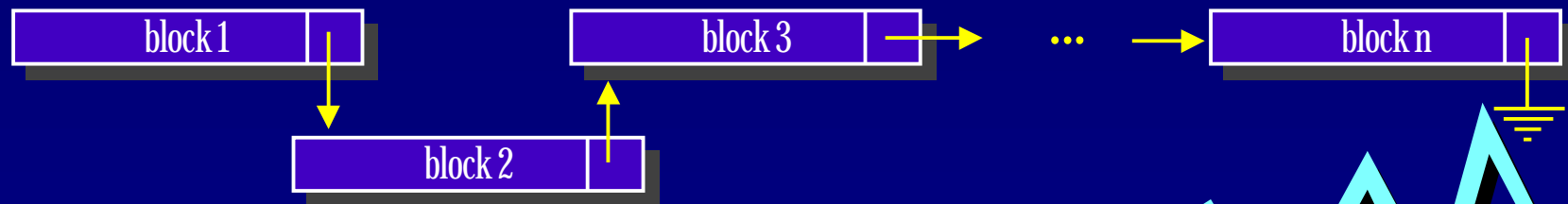
# Physical Linked Sequential Files

## ▼ Physical sequential files

- *InsertOne* & *DeleteOne* cause file to be cluttered

## ▼ Physical linked sequential files

- Block pointers preserve block adjacency.
- Blocks could be dispersed on the disk.
- Each block access is a random access.
- Ok for highly active files.



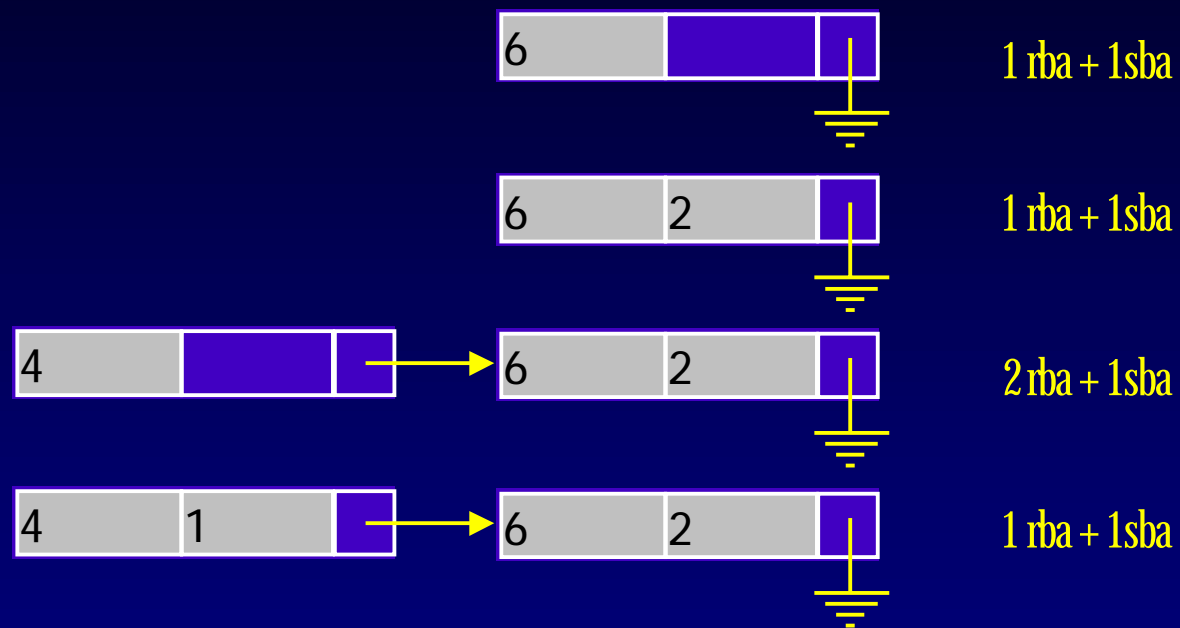
# Physical Linked Sequential Files

▼ Binary search can not be done.

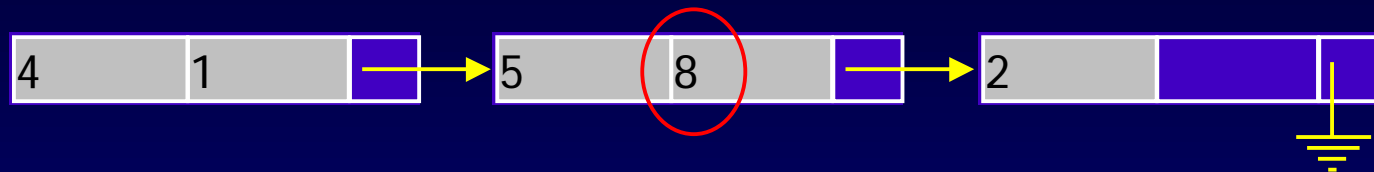
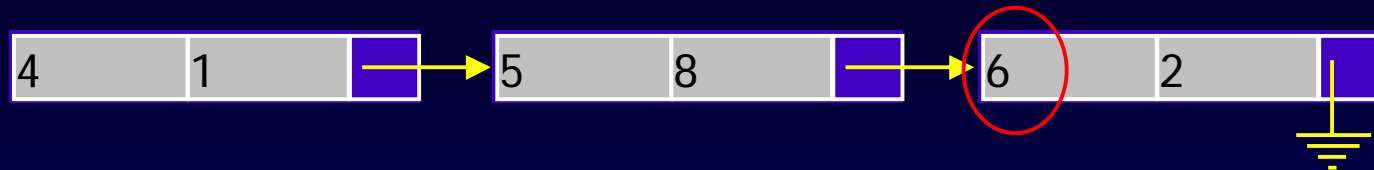
	Unordered	Ordered
<i>RetrieveAll</i>	$NBLK \ rba$	$NBLK \ rba$
<i>RetrieveOne</i> (found)	$(1+NBLK) / 2 \ rba$	$(1+NBLK) / 2 \ rba$
<i>RetrieveOne</i> (not found)	$NBLK \ rba$	$(1+NBLK) / 2 \ rba$



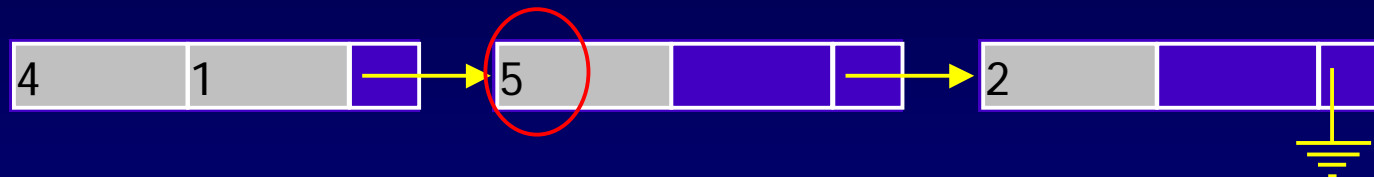
# Unordered Files : *InsertOne*



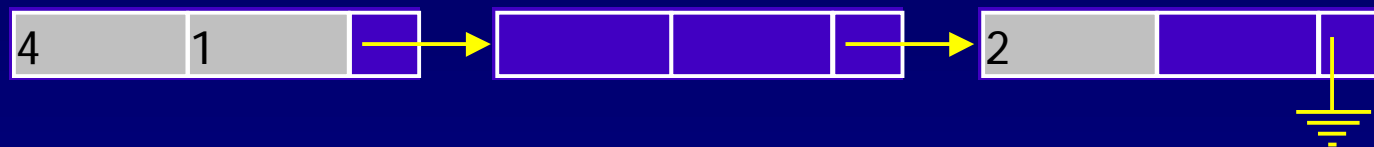
# Unordered Files : *DeleteOne*



*RetrieveOne* + 1 sba

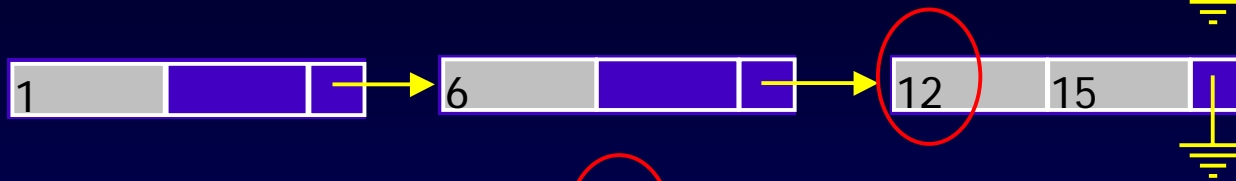


*RetrieveOne* + 1 sba

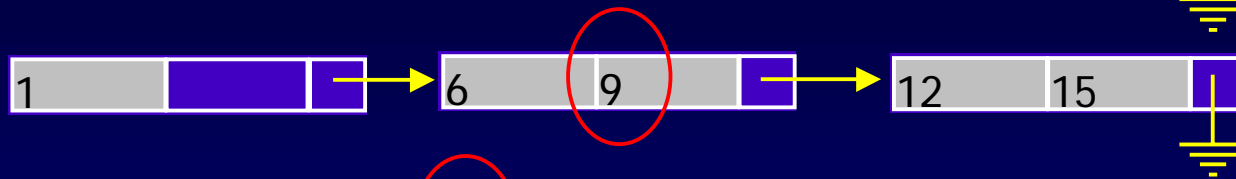


*RetrieveOne* + 1 sba

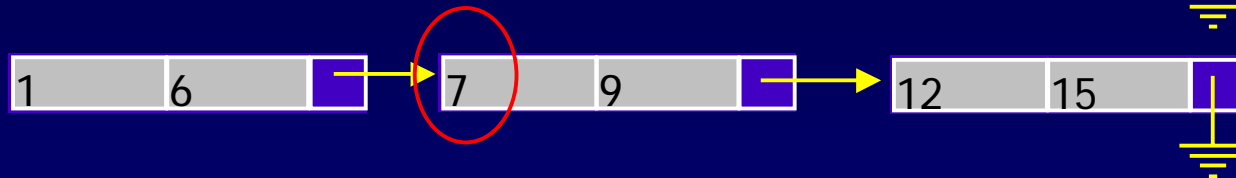
# Ordered Files : *InsertOne*



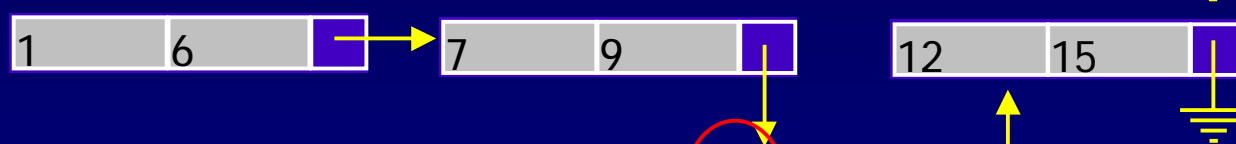
*RetrieveOne* + 1 sba



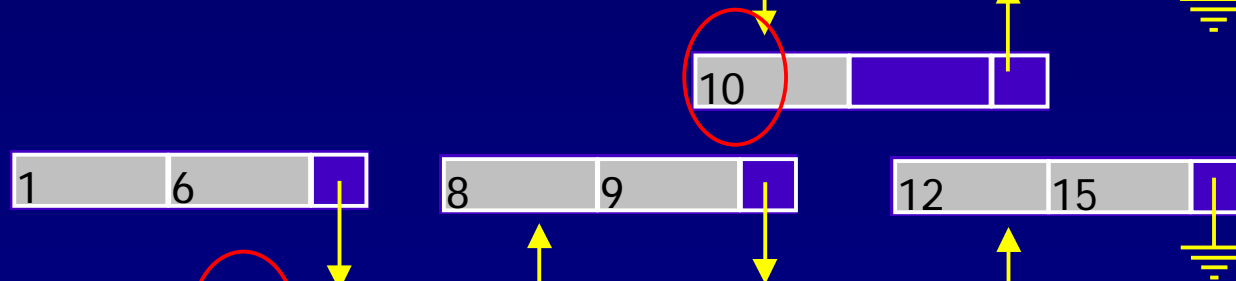
*RetrieveOne* + 1 rba + 1 sba



*RetrieveOne* + 1 rba + 2 sba



*RetrieveOne* + 2 rba + 2 sba



*RetrieveOne* + 2 rba + 3 sba

# Sequential Files in COBOL

FILE SECTION.

FD OLD-MASTER

LABEL RECORDS ARE STANDARD  
BLOCK CONTAINS 20 RECORDS

01 OLD-MASTER-REC.

05 M-ACCT-NO PIC X(5).

05 AMOUNT-DUE PIC 9(4)V99.

05 FILLER PIC X(89).

...

OPEN INPUT OLD-MASTER TRANS-FILE  
OUTPUT NEW-MASTER.

READ OLD-MAS-REC AT END MOVE HIGH-VALUES TO M-ACCT-NO.

...

CLOSE OLD-MASTER TRANSF-FILE NEW-MASTER.