USING INHERITANCE FOR RECORD BUFFER CLASSES

4.3 Using Inheritance for Record Buffer Classes

- **Inheritance**
  The implicit inclusion of members of a parent class in a child class.

Delineation of Records in a File

fixed length record

A record which is predetermined to be the same length as the other records in the file.

Record 1	Record 2	Record 3	Record 4	Record 5

- The file is divided into records of equal size.
- All records within a file have the same size.
- Different files can have different length records.
- Programs which access the file must know the record length.
- Offset, or position, of the nth record of a file can be calculated.
- There is no external overhead for record separation.
- There may be internal fragmentation (unused space within records.)
- There will be no external fragmentation (unused space outside of records) except for deleted records.
- Individual records can always be updated in place.
- Example (80 byte records):
- 0 66 69 72 73 74 20 6C 69 6E 65 0 0 1 0 0 0 first line.....
- 10 0 0 0 0 0 0 0 0 FF FF FF FF 0 0 0 0 .....
- 20 68 FB 12 0 DC E0 40 0 3C BA 42 0 78 FB 12 0 h.....@.<.B.x...
- 30 CD E3 40 0 3C BA 42 0 8 BB 42 0 E4 FB 12 0 ..@.<.B...B.....
- 40 3C 18 41 0 C4 FB 12 0 2 0 0 0 FC 3A 7C 0 <.A.....:|.
- 50 73 65 63 6F 6E 64 20 6C 69 6E 65 0 1 0 0 0 second line.....
- 60 0 0 0 0 0 0 0 0 FF FF FF FF 0 0 0 0 .....
- 70 68 FB 12 0 DC E0 40 0 3C BA 42 0 78 FB 12 0 h.....@.<.B.x...
- 80 CD E3 40 0 3C BA 42 0 8 BB 42 0 E4 FB 12 0 ..@.<.B...B.....
- 90 3C 18 41 0 C4 FB 12 0 2 0 0 0 FC 3A 7C 0 <.A.....:|.
- Advantage: the offset of each record can be calculated from its record number. This makes direct access possible.
- Advantage: there is no space overhead.
- Disadvantage: there will probably be internal fragmentation (unusable space within records.)

Delimited Variable Length Records

variable length record

A record which can differ in length from the other records of the file.

delimited record

A variable length record which is terminated by a special character or sequence of characters.

delimiter

A special character or group of characters stored after a field or record, which indicates the end of the preceding unit.

Record 1	#	**Record 2**	#	**Record 3**	#	**Record 4**	#	**Record 5**	#

- The records within a file are followed by a delimiting byte or series of bytes.
- The delimiter cannot occur within the records.
- Records within a file can have different sizes.

- Different files can have different length records.
- Programs which access the file must know the delimiter.
- Offset, or position, of the nth record of a file cannot be calculated.
- There is external overhead for record separation equal to the size of the delimiter per record.
- There should be no internal fragmentation (unused space within records.)
- There may be no external fragmentation (unused space outside of records) after file updating.
- Individual records cannot always be updated in place.
- Algorithms for Accessing Delimited Variable Length Records
- Code for Accessing Delimited Variable Length Records
- Code for Accessing Variable Length Line Records
- Example (Delimiter = ASCII 30 (IE) = RS character:
- 0 66 69 72 73 74 20 6C 69 6E 65 1E 73 65 63 6F 6E first line.secon
- 10 64 20 6C 69 6E 65 1E d line.
- Example (Delimiter = '\n'):
- 0 46 69 72 73 74 20 28 31 73 74 29 20 4C 69 6E 65 First (1st) Line
- 10 D A 53 65 63 6F 6E 64 20 28 32 6E 64 29 20 6C ..Second (2nd) l
- 20 69 6E 65 D A ine..

- Disadvantage: the offset of each record cannot be calculated from its record number. This makes direct access impossible.
- Advantage: there is space overhead for the length prefix.
- Advantage: there will probably be no internal fragmentation (unusable space within records.)