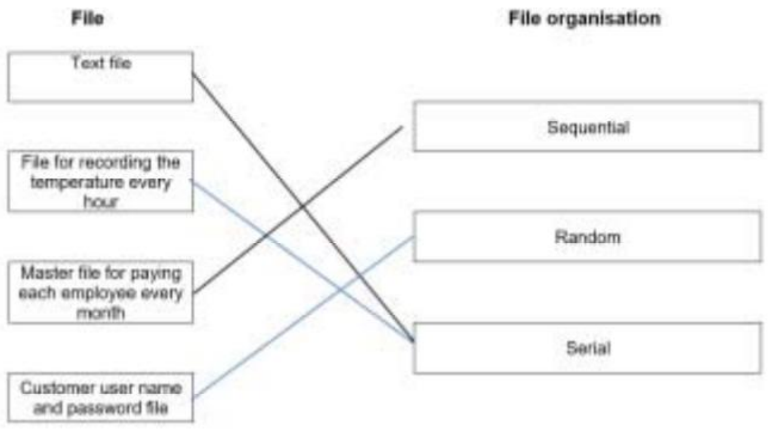
Answer

Answer 1

2	**One** mark for each correct line drawn  File File organisation Text file File for recording the temperature every hour Master file for paying each employee every month Customer user name and password file Sequential Random Serial	**4**

Answer 2

5(b)(i)	1 mark per bullet point to **max 2** • So the readings are stored in **chronological** order • Easy to add / append each new reading to the end of the file // no further processing is required • Allows the readings to be read in the order that they were taken • Readings do not need to be given further identification as to date / time // no key field needs to be added	**2**
5(b)(ii)	1 mark per bullet point **max 2** • Earliest temperature reading is accessed first • and each successive temperature reading is read (in date / time order) • until the final reading has been accessed	**2**
5(b)(iii)	1 mark for Random 1 mark per bullet point for description to **max 3** • Record locations are calculated • … using a hashing algorithm **on a key field** • If a record cannot be stored / found at that location • … then subsequent locations are searched // closed hash • ... or an overflow area is searched // open hash	**4**

Answer 3

4(a)	Example: Speed of access Just used as a look-up file No need for any serial or sequential processing 1 mark for any valid point	1

4(b)(i) — 1

CustomerID	RecordKey
802139	2139
700004	4
689998	89998
102139	2139

4(b)(ii)	Minimum value: 0 Maximum value: 99999	1 1	2

4(b)(iii) — 4

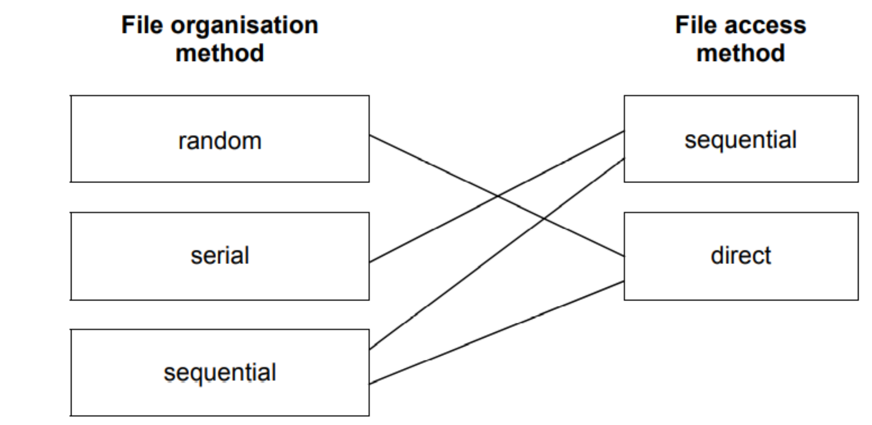
```
PROCEDURE InsertRecord(CustomerID : INTEGER)
   RecordKey ← CustomerID MOD 100000
   Success ← FALSE
   // Find position for new record and insert it
   REPEAT
      IF record at position RecordKey is empty
         THEN
            Insert new record at position RecordKey
            Success ← TRUE
         ELSE
            IF RecordKey = 99999
               THEN
                  RecordKey ← 0
               ELSE
                  RecordKey ← RecordKey + 1
            ENDIF
      ENDIF
   UNTIL Success = TRUE
ENDPROCEDURE
```

4(c)(i)	For security If file is hacked then encrypted PIN cannot be used Only encrypted PINs are transmitted and compared 1 mark for any valid point	Max 2

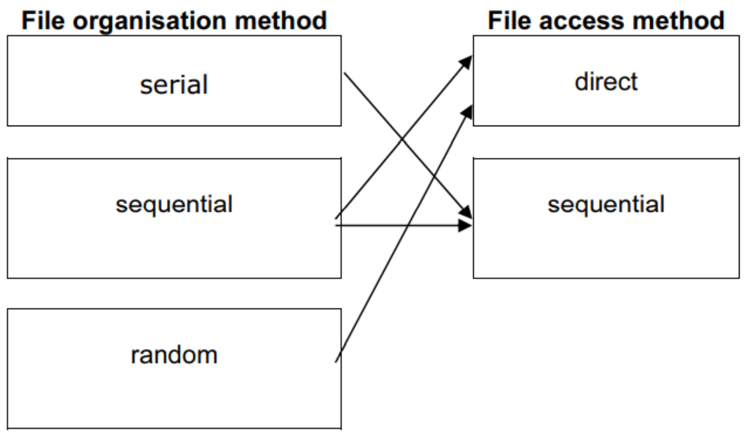
4(c)(ii) — 3

```
1. Customer ID is read from card
2. Customer enters PIN
3. Customer PIN is encrypted
4. Customer ID is hashed
5. Customer record is located in file
6. PIN is checked against PIN in record
7. If match then transaction can proceed
```

Answer 4

4(a)	 File organisation method File access method random serial sequential sequential direct 1 mark for random correct 1 mark for serial correct 2 marks for sequential correct (1 per correct line)	4
4(b)(i)	File A: Serial 1 Meter readings are submitted over time // added to the end of file 1 Stored chronologically 1	3
4(b)(ii)	File B: Sequential 1 Any two points from: Each customer has a unique account number 1 Sorted on Account number 1 High hit rate // Suitable for batch processing monthly statements 1	3
4(b)(iii)	File C: Random 1 Login without waiting // Random organisation allows fastest direct access to required record 1 Low hit rate // Suitable for access to individual records 1	3

Answer 5

4 (a)	 File organisation method File access method serial sequential random direct sequential	**1** **2** **1**

(b) (i)	Sequential As all customers get statement … // high hit rate Suitable for batch processing of the records // the records will be processed one after the other File organised using customer's unique ID (as primary key field) // Serial As all customers get statement … // high hit rate Suitable for batch processing of the records // the records will be processed one after the other Order not important	**1** **1** **1** **1** **1** **1** **1** **1** **Max 3**
(ii)	Random Real-time transaction processing Requires fastest access to data No need to search through records	**1** **1** **1** **1** **Max 3**
(iii)	Serial Each new record is appended Transactions are recorded in chronological order File re-organisation not required for each new record // no need for the records to be sorted	**1** **1** **1** **Max 3**

Answer 6

(ii)	• no need to re-sort data every time new data is added • only a small file so searching will require little processing • new records can easily be appended	1 1 1 [max 2]

Answer 7

(ii)	`StationID` is hashed to produce home location If home location is free insert record Else use overflow method to find free location	1 1 1