Mark scheme

Chapter1

Section A

1. Failover system / redundancy / removable data / offsite storage / online storage(cloud)

|  |  |
| --- | --- |
| Alpha testing | Beta testing |
| Alpha testing performed by testers who are usually internal employees of the organization | Beta testing is performed by clients or end users who are not employees of the organizations |
| Alpha testing performed at developer’s site | Beta testing is performed at client location or end user of the product |

1. Incompatible file formats / data structure differences / validation rules / incomplete data transfers / international conventions on dates, currencies & character sets

Section B

1. User documentation: help files / online support / printed manuals

Simpler is better & quality is important

1. Online storage (cloud)

Causes of data loss: hardware/system malfunctions / human error / software corruption / malicious software(viruses) / natural disasters

1. User may be unsatisfied with system / system may be unsuited for user’s problem, affecting productivity
2. Phased conversion

Advantages: allows people to get used to the new system; training of staff can be done in stages

Disadvantages: if the new system fails, there is no fall back for that part of the system

Chapter4

Section A

1. Abstraction can be viewed both as process and as an entity

Enables a person to concentrate on the essential aspects of the problem on hand, while ignoring details that tend to be distracting.

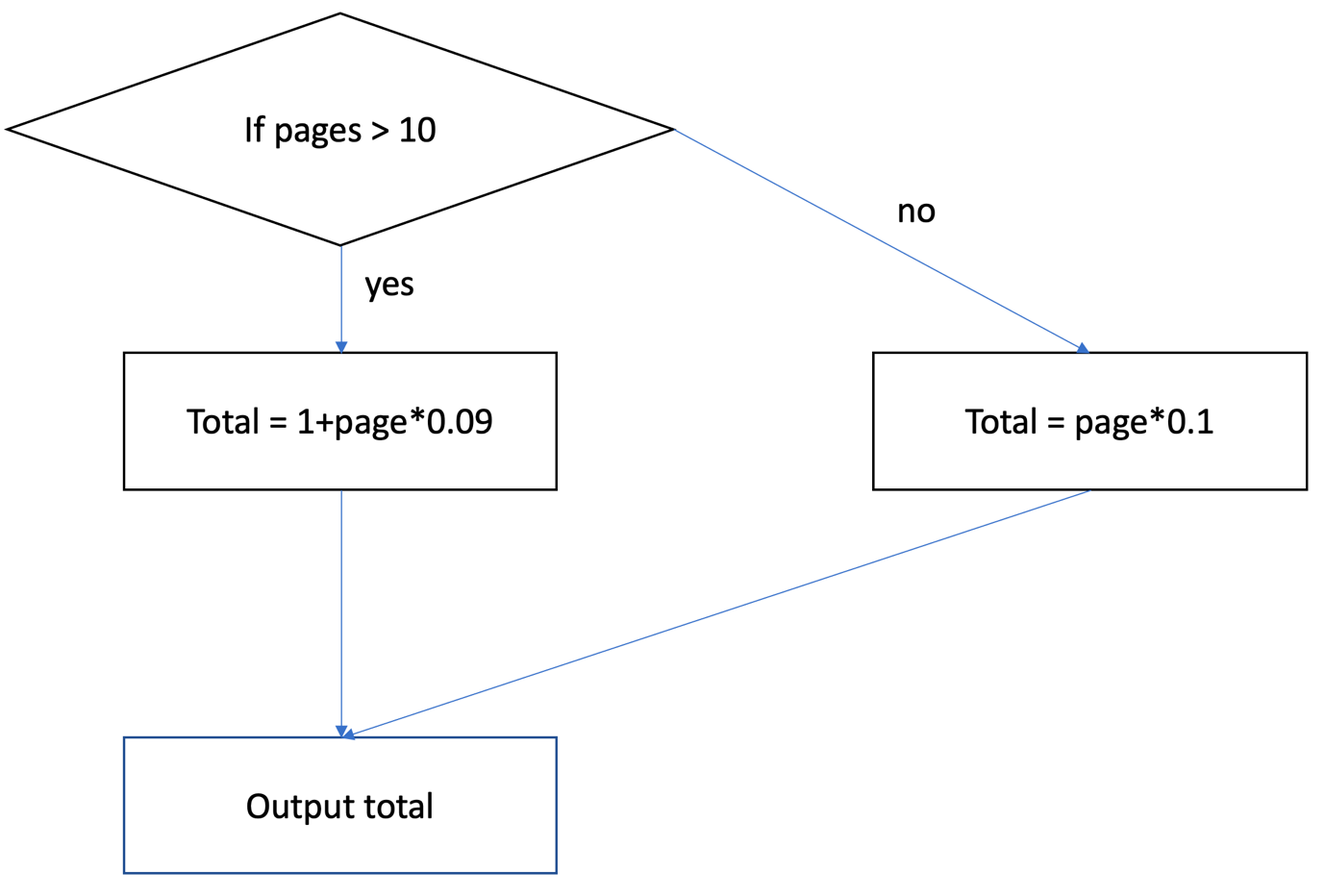
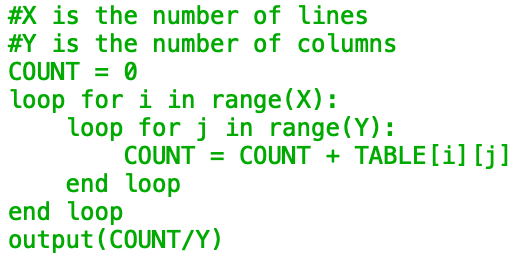
Abstraction is a convenient way to deal with complexity.

1. Increase computation speed / increase complexity of programming language and hardware / reduces complexity of working with array operations within loops, of performing matrix multiplication, of conducting parallel searches in databases.
2. Pre-planning / pre-conditions / identify inputs and outputs required in a solution
3. List
4. 
5. 
6. Fib(5) = Fib(4) + fib(3)

= fib(3) + fib(2) + fib(2) + fib(1)

= fib(2) + fib(1) + fib(1) + fib(0) …..

**3**

1. They should specify program objectives ; program users/collect the information from the users; output requirements; input requirements; processing requirements; document the requirements and objectives
2. 
3. 
4. 
5. 



Chapter 5

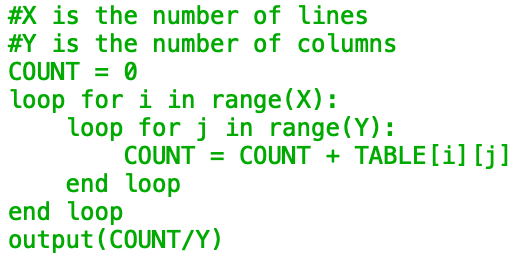
Section A

1. A linked list is a linear collection of self-referential structures, called nodes, connected by pointer links. Subsequent nodes are accessed via a link pointer number that is stored in each node.
2. Parents: A, B, C, E

Roots: A

Left children: B, D, E, G

Section B

1. Two-dimensional array
2. 
3. 
4. Each node would hold the data for one train (ID, Departure, Destination, ARRIVED);

Head pointer points to the first in the list

Each subsequent pointer points to the next in the list and last node has null pointer

1. This is a really good question
2. Picture
3. Stack: first in first out

Elements in a linked list could be removed from any position in the list; hence a linked list is better;