NAME: Sayed Muslim shah

ID#: 14856

MODULE: Bachelors {Software Engineering}

SUBJECT: Software Engineering

SEMESTER: 4

SECTION: B

INSTRUCTOR: Engr. Ghassan Husnain

DATE: 22/06/2020

Question 1: Case Description

INU Printing Press, a part of a University undertakes printing jobs from different departments of the University. The press can take printing related orders from other organizations or individuals also, but this is a rare practice. Orders once placed cannot be canceled. Any financial transactions of the various departments with the printing press are done through the University Office only. The University Office maintains the accounts of the press and the various departments of the University. The press acquires its inventory from outside suppliers. This is done by accepting the quotations from various suppliers and placing orders respectively. They maintain information of vendors / suppliers for the raw materials. A supplier can supply more than one item and an item can be supplied by more than one supplier. The supplier may supply the item in parts. Stock for a particular item can be acquired more than once in a year. The suppliers send two copies of invoice to the printing press, of which one is forwarded to the University Office. The press makes payment to the suppliers after obtaining approval from the University Office. The department places printing related order with the press and depending on the inventory, the press is entitled to give a response specifying whether the order will be fulfilled immediately or in the near future. Each department is assigned a budget at the beginning of the year, which becomes an important factor while getting any concession or discount from the press. Once the order is accepted, it has to go through various printing processes for its completion. However, these processes or the sequencing are not mandatory as a job or task may need different processing or handling. The various processes are:

- Composing
- Pasting
- Plate making
- Printing
- Binding

At the completion of the first copy, it is sent to the department for approval, which when received initiates the completion of the order. For the completion of the order, the press

maintains a document known as costing sheet which specifies what all processes were applied and the inventory used for the completion of a particular order. This document is important in generating the bill.

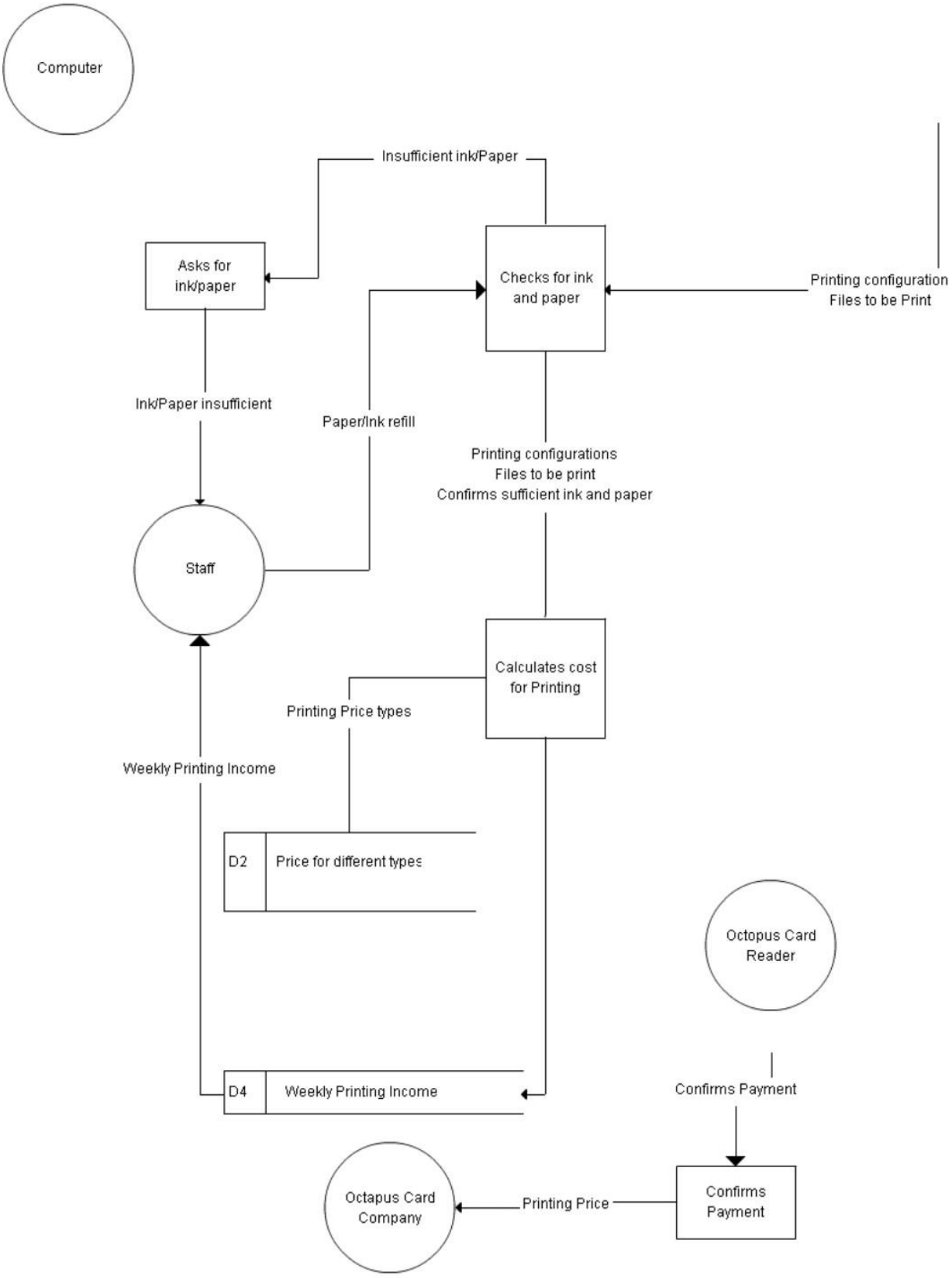
3 delivery challans are sent along with the items; one being kept in the press, second being given to the department and the third being sent to the University Office by the press. The invoice is not sent with the items, but it is sent at a later date due to complexity involved in preparing it. Two invoices are sent to the department; of which one is forwarded to the University Office by the department.

Summary of transactions as well as reports related to the printing press, are sent to the University Office on a regular basis. These reports or summary is useful to the management for analyzing their current budget and planning accordingly for future budget.

Q.1.1: Draw a Context diagram for INU Printing Press?

ANSWER:

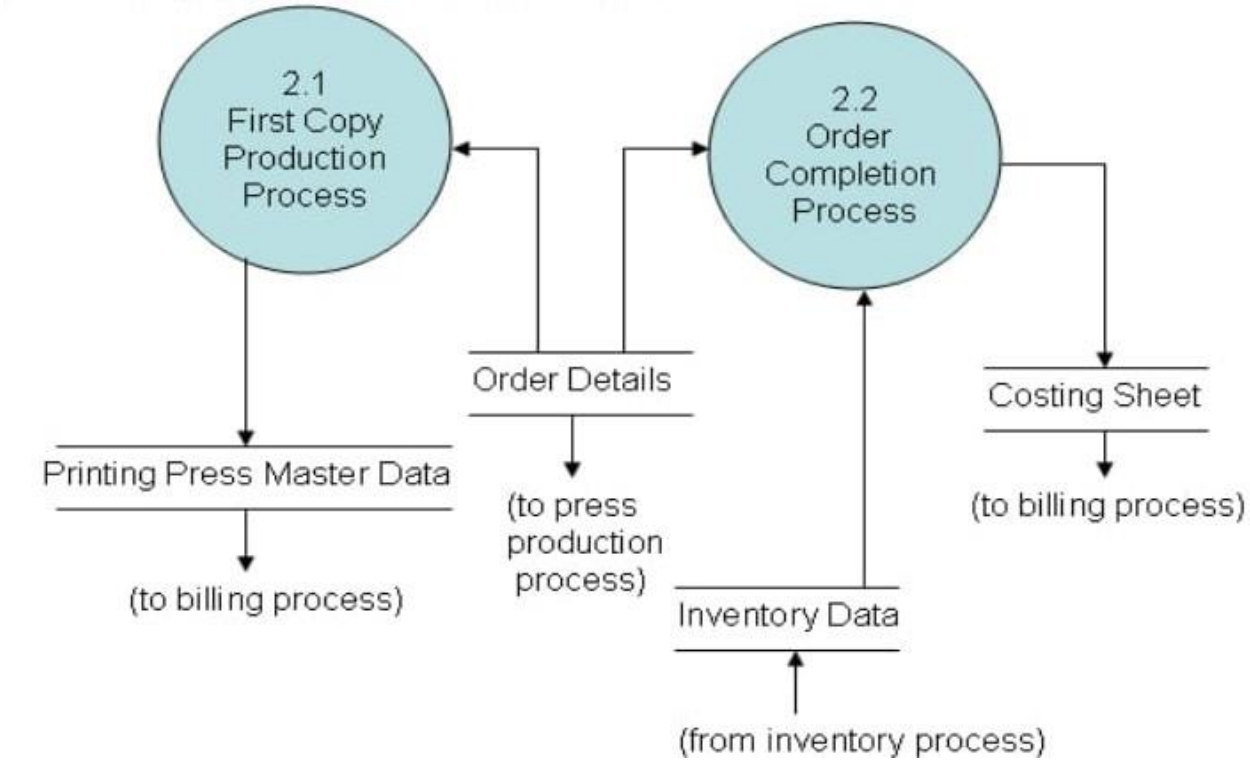
Context diagram:


Computer
Insufficient ink/Paper
Asks for ink/paper
Checks for ink and paper
Printing configuration
Files to be Print
Ink/Paper insufficient
Paper/Ink refill
Printing configurations
Files to be print
Confirms sufficient ink and paper
Staff
Calculates cost for Printing
Printing Price types
Weekly Printing Income
D2
Price for different types
Octopus Card Reader
D4
Weekly Printing Income
Confirms Payment
Octapus Card Company
Printing Price
Confirms Payment

Q.1.2: Draw a Level 1 Data Flow Diagram (DFD) for the above case study?

ANSWER:

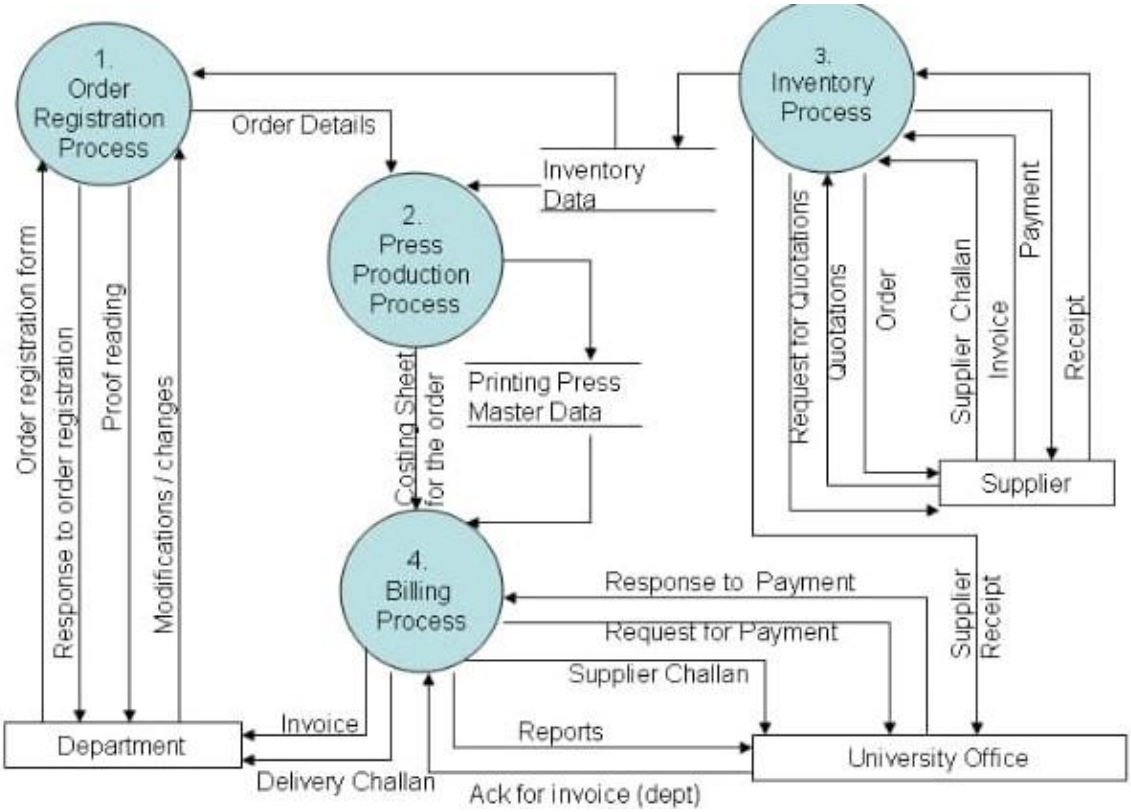
Level 1 Data Flow Diagram (DFD):



Q.1.3: Draw a Level 2 DFD for the Order Registration Process, Press Production Process, Inventory Process, and Billing Process?

ANSWER:

Level 2 DFD for the Order Registration Process, Press Production Process, Inventory Process, and Billing Process:



Question 2:

Q.2.1: Explain why testing can only detect the presence of errors, not their absence?

ANSWER:

Testing can detect only the presence of errors, not their absence because the main goal of the testing is:

to observe the behavior of the particular software and to check whether it meet its requirement expectation or not.

Testing is a part of broader process of software verification and validation. It consists of a set of activities, where the testers try to make the software behave anomalous in order to detect or anomaly to be later fix. Testing cannot demonstrate the faults other than specified in every circumstance. It is always possible that a test have overlooked could discover further problem with the system.

Q.2.2: Define the following terms:

1. **Unit Testing**

ANSWER:

UNIT TESTING is a level of software testing where individual units/ components of a software are tested. The purpose is to validate that each unit of the software performs as designed. A unit is the smallest testable part of any software. It usually has one or a few inputs and usually a single output.

- The most 'micro' scale of testing.
- Tests done on particular functions or code modules.
- Requires knowledge of the internal program design and code.
- Done by Programmers (not by testers).

2. **System Testing**

ANSWER:

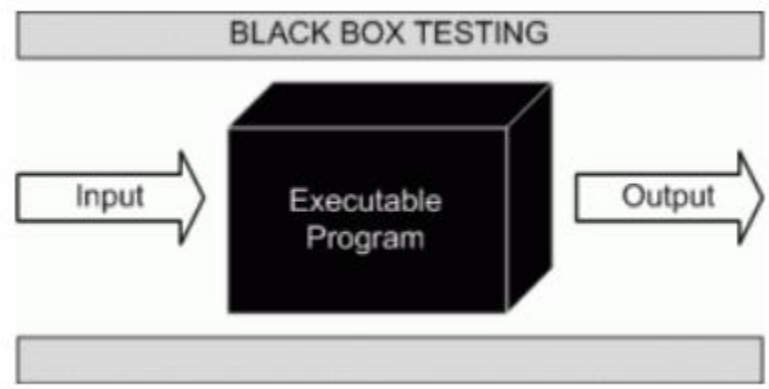
SYSTEM TESTING is a level of testing that validates the complete and fully integrated software product. The purpose of a system test is to evaluate the end-to-end system specifications. Usually, the software is only one element of a larger computer-based system.

3. Black Box Testing

ANSWER:

BLACK BOX TESTING, also known as Behavioral Testing, is a software_testing method in which the internal structure/design/implementation of the item being tested is not known to the tester. These tests can be functional or non-functional, though usually functional.

- No knowledge of internal design or code required.
- Tests are based on requirements and functionality
- Not based on any knowledge of internal design or code
- Covers all combined parts of a system
- Tests are data driven (Tests are based on putting some data to check the system)
- It uncovers:
  - Incorrect or missing functions
  - Interface errors
  - Errors in data structures or external database access
  - Performance errors
  - Initialization and termination errors



4. **White Box Testing**

ANSWER:

White box testing / Structural testing:

- Based on knowledge of internal logic of an application's code
- Based on coverage of code statements, branches, paths, conditions
- Tests are logic driven
- It ensures
  - All independent paths within a module have been exercised at least once
  - Exercise all logical decisions on their *true* and *false* sides
  - Execute all loops at their boundaries and within their operational bounds
  - Exercise internal data structures to ensure their validity

Question 3:

Q.3.1: Briefly describe the three main types of software maintenance. Why is it sometimes difficult to distinguish between them?

ANSWER:
The three main types of software maintenance are as follows:

1. **Fault repairs:** Coding errors are usually relatively cheap to correct; design errors are more expensive as they may involve rewriting several program components. Requirements errors are the most expensive to repair because of the expensive system redesign which be necessary.

2. **Environmental adaptation:** This type of maintenance is required when some aspect of the system's environment such as the hardware, the platform operating system, or other support software changes the application system must be modified to adapt it to cope with these environmental changes.

3. **Functionality addition:** This type of maintenance is necessary when the system requirements change in response to organizational or business change. the scale of the changes required to the software is often much greater than for the other types of maintenance.

Why is it difficult to differentiate between the types of maintenance?

In practice, there is not a clear-cut distinction between these types of maintenance, when the system adapts to new environment, then add functionality to take advantage of new environmental features. Software faults are often exposed because users use the system in unanticipated ways. These types of maintenance are recognized but a different person sometimes gives them different names.

‘Corrective maintenance’ is universally used to refer to maintenance for fault repair’

“Adaptive maintenance’ sometimes means adapting to new environment and sometimes means adapting the software to new requirements.

‘Perfective maintenance’ sometimes means perfecting the software by implementing new requirements; in other cases, it means maintaining the functionality of the system but improving its structure and performance.

Q.3.2: What are the principal factors that affect the costs of system reengineering? Also, briefly explain the reengineering process with the help of diagram:

ANSWER:

Software Re-Engineering is the examination and alteration of a system to reconstitute it in a new form. The principles of Re-Engineering when applied to the software development process is called software re-engineering. It affects positively at software cost, quality, service to the customer and speed of delivery. In Software Re-engineering, we are improving the software to make it more efficient and effective.

Re-Engineering cost factors:

- The quality of the software to be re-engineered.
- The tool support availability for engineering.
- Extent of the data conversion which is required.
- The availability of expert staff for Re-engineering.

Reengineering process activities:

Source code translation

1)Convert code to a new language.

Reverse engineering

2)Analyze the program to understand it;

Program structure improvement

3)Restructure automatically for understandability;

Program modularization

4)Reorganize the program structure;

Data reengineering

5)Clean-up and restructure system data.

Diagram: