



# Software Engineering

هندسة البرمجيات

جامعة بغداد

كلية التربية للعلوم الصرفة/ابن

الهيثم

قسم علوم الحاسبات

المرحلة الثالثة

م.د. علي يحيى غني



# نظري محاضرة الاسبوع العثرون

## Software Design Strategies

# Topics covered

---



- ✧ Structured Design
- ✧ Function Oriented Design
- ✧ Object Oriented Design
- ✧ Software Interface Design

# Structured Design



- ✧ **Structured design** is a conceptualization of problem into several well-organized elements of solution. It is basically concerned with the solution design. **Benefit of structured design** is, it gives **better understanding** of how the problem is being solved. Structured design also makes it **simpler** for designer to concentrate on the problem more accurately.
- ✧ Structured design is mostly based on '**divide and conquer(Control)**' strategy where a problem is broken into several small problems and each small problem is individually solved until the whole problem is solved.

# Structured Design



- ✧ The small pieces of problem are solved by means of solution **modules**. Structure design emphasis that these modules be well organized in order to achieve precise solution.
- ✧ These modules communicate with each other. **A good structured design** always follows some rules for communication among multiple modules, namely:
  - ❑ **Cohesion** - grouping of all functionally related elements.
  - ❑ **Coupling** - communication between different modules.
    - ✧ A good structured design has high cohesion and low coupling arrangements.

# Structured Design



- ✧ **Structured programming:** is a style of programming designed to make programs more comprehensible and programming errors less frequent. Structured programming should usually includes:
1. **Block structure:** The statements in the program must be organized into functional groups.
  2. **Avoidance of jumps:** (“GO-TO-less programming”)
  3. **Modularity:** Programs should be Brocken up into subroutines, even if some of the subroutines are called only one.

# Structured Design



## ✧ Note:

In different programming languages, a **subroutine** may be called a **routine**, **subprogram**, **function**, **method**, or **procedure**. Technically, these terms all have different definitions.

# Structured Design



## Structured Pascal program

```
IF x<=y THEN
  BEGIN
    z := y - x;
    q := SQRT (Z);
  END
ELSE
  BEGIN
    z:= x-y;
    q:= -sqrt (z)
  END;
WRITELEN (z , q);
```

## Unstructured Pascal program

```
IF x>y THEN GOTO 2;
Z  := Y - X;
q  := sqrt (z);
GOTO 1;
2: Z:= X - Y;
q:= -sqrt (z);
1:= writeln (z , q);
```

# Function Oriented Design



- ✧ In **function oriented design**, the system comprises of many smaller sub-systems known as functions. These functions are capable of performing significant task in the system. The system is considered as top view of all functions.
- ✧ This design mechanism divides the whole system into **smaller functions**, which provides means of abstraction by concealing the information and their operation. These functional modules can share information among themselves by means of information passing and using information available globally.

# Function Oriented Design (Vending Machine)



Figure 32: Vending Machine

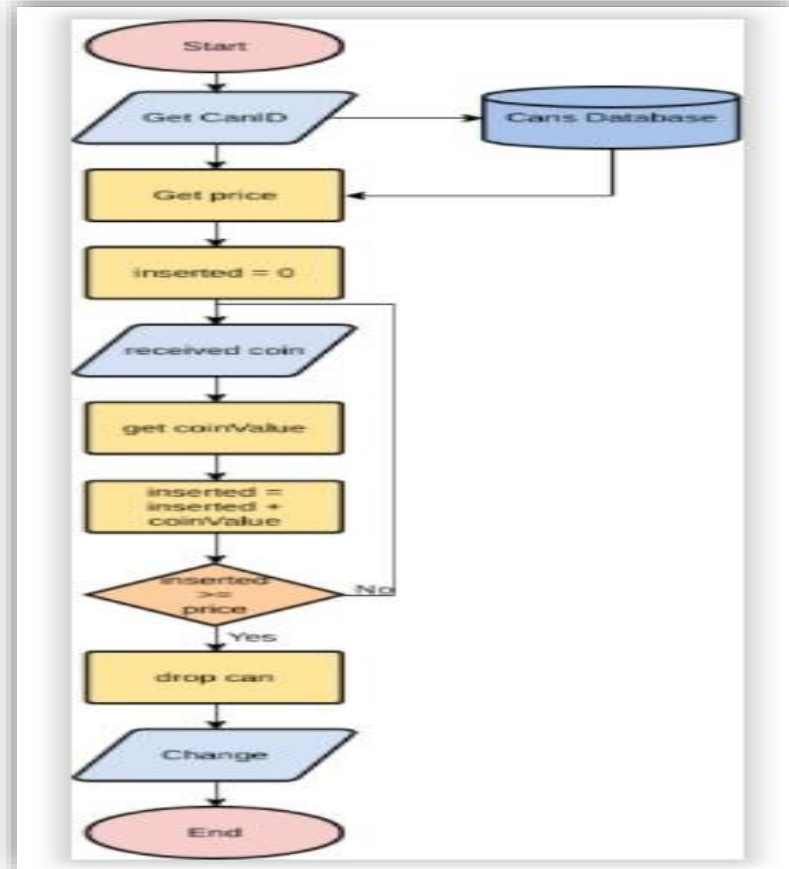


Figure 33: DFD of Vending Machine

# Function Oriented Design (ATM)

## ATM machine Functions

- Cash Withdraw
- Deposit
- Transfer
- Balance inquiry
- Mobile Top-Up

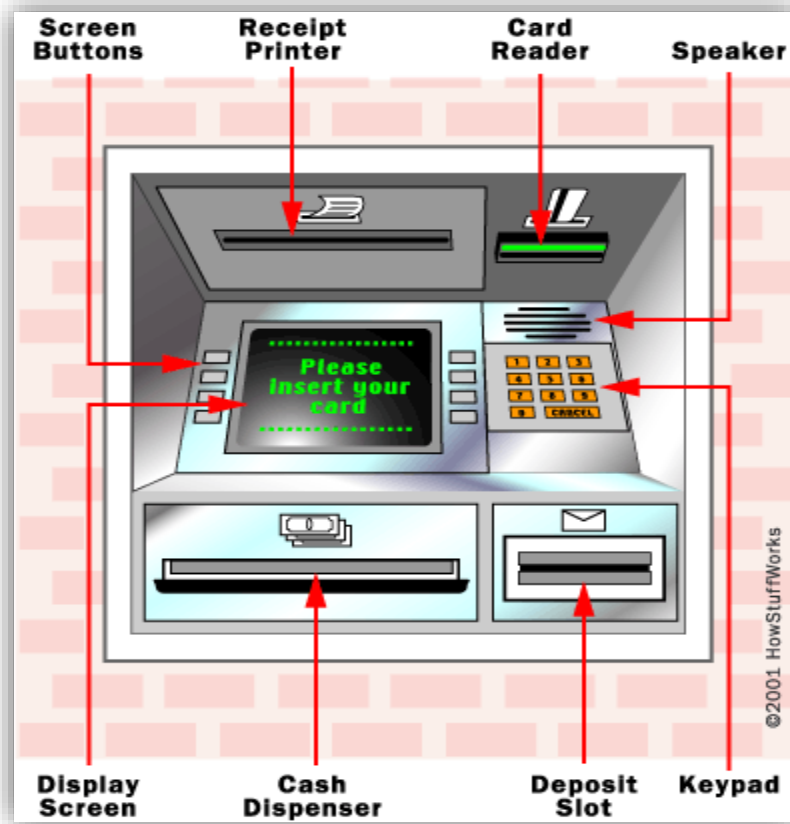


Figure 35: ATM Machine

## Report (2)



- As a software designer, What are the main functions and sub functions need to consider to design ATM Machine (Function Oriented Design) for a new bank? You can use some design tools such as:
- DFD



# Thanks