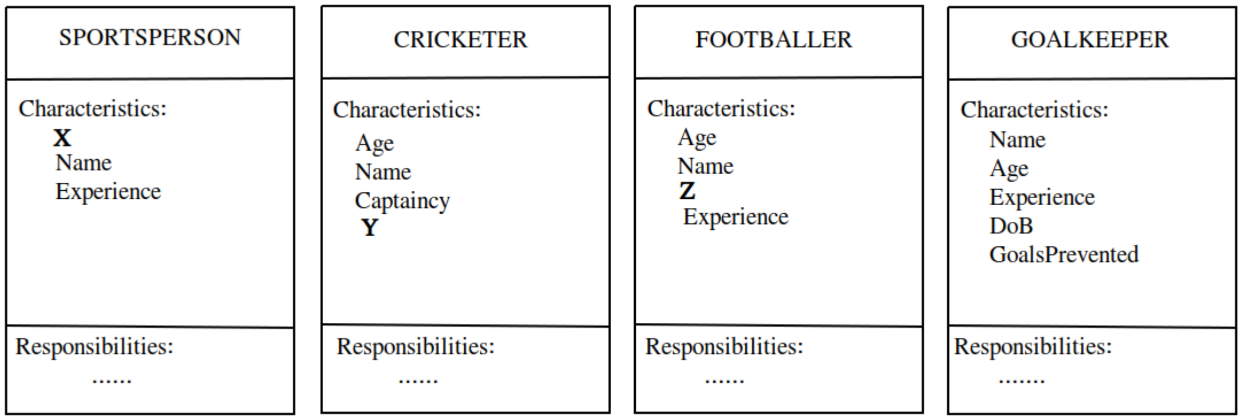
Object Oriented Analysis and Design: Assignment 2

Total Marks : 20

July 11, 2022

Question 1

Consider the following abstractions :



Arrange these into a natural abstraction hierarchy to find the values of `X`, `Y`, and `Z`.

Marks: 2 **MCQ**

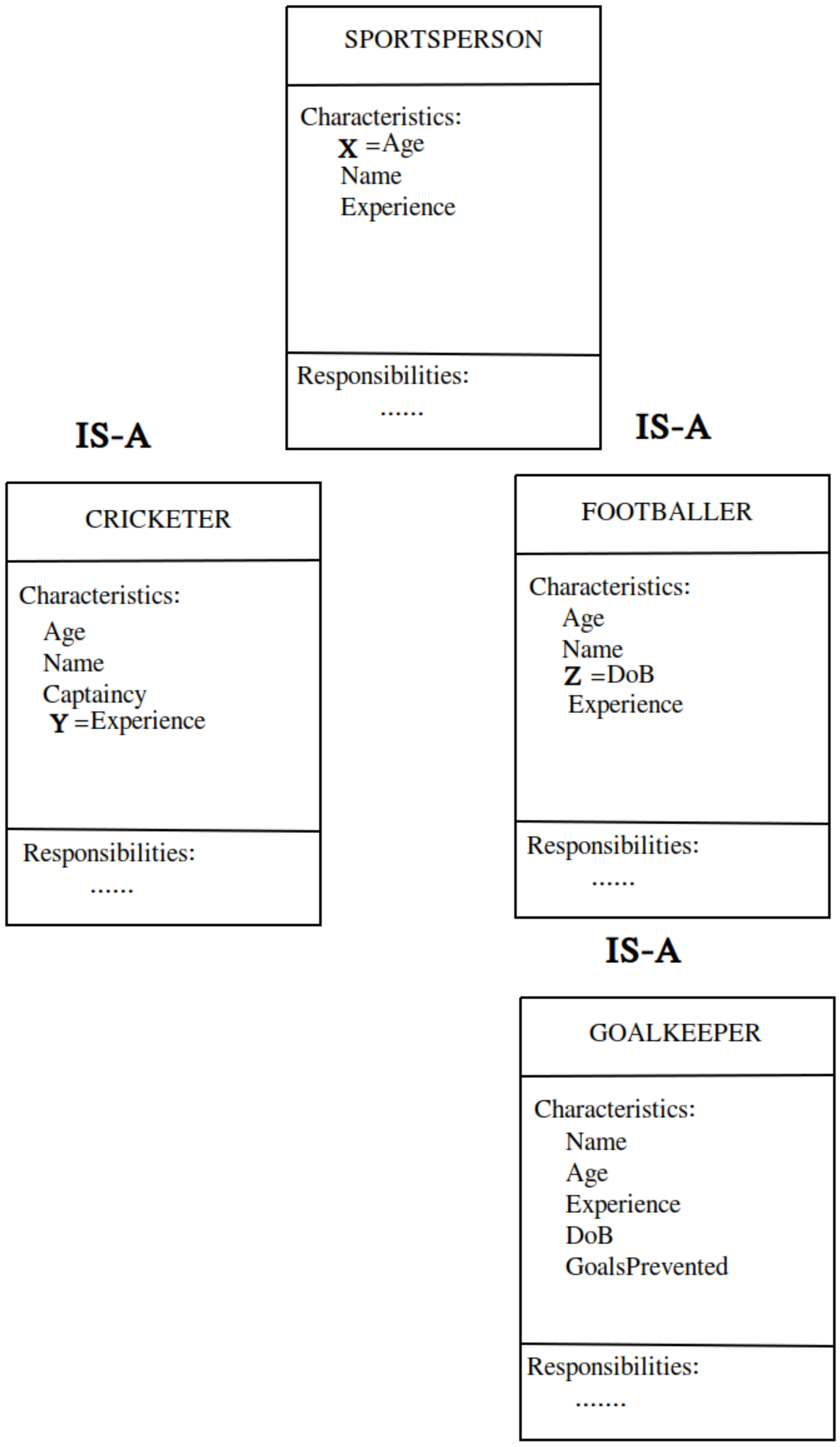
a) `X=Captaincy, Y=Experience, Z=GoalsPrevented`

b) `X=Age, Y=DoB, Z=GoalsPrevented`

c) `X=Age, Y=Experience, Z=DoB`

d) `X=Captaincy, Y=DoB, Z=Age`

Answer: c)
Explanation: The natural hierarchy should be as follows



Since `CRICKETER` and `FOOTBALLER` are both sportsman, they should share the characteristics of `SPORTSPERSON`. The given characteristics of `SPORTSPERSON` are `X, Name, Experience`. `Experience` is not present in the characteristics of `CRICKETER`. Thus, `Y=Experience`. The only other common characteristic between `CRICKETER` and `FOOTBALLER` is `Age`. Hence, `X=Age`. From the given characteristics of `GOALKEEPER`, `Z=DoB` or `Z=GoalsPrevented`. However, `GoalsPrevented` is generally specifically related to `GOALKEEPER`. Also, from the given options, `Z=DoB`. Hence, option (c) is correct.

Question 2

Consider the following two abstraction hierarchy systems :
System1: A ***`Service Agent`*** *has a Name and is responsible for requesting the customer's service number. A* ***`Delivery Service Agent`*** *is a* ***`Service Agent`*** *who is additionally characterised by a Company and delivery type. They are additionally responsible for recording the estimated delivery date of the item.* ***`Paint Service Agent`*** *is also a* ***`Service Agent`*** *who additionally has a Service Code. The* ***`Paint Service Agents`*** *are also responsible for generating a demo paint service code.*
System2: A ***`Service Agent`*** *has a Name and is responsible for requesting the customer's service number. A* ***`Delivery Service Agent`*** *is a* ***`Service Agent`*** *who is additionally characterised by a Company and delivery type. They are additionally responsible for recording the estimated delivery date of the item and generating a new service request code.* ***`Paint Service Agent`*** *is also a* ***`Service Agent`*** *who additionally has a Company and Paint Code. The* ***`Paint Service Agents`*** *are also responsible for generating a demo paint service code and a new service request code.*

Which of the following options is true if `System 1` and `System 2` are to be developed into two different abstraction hierarchies? *Marks: 2* **MCQ**

`a) An additional level of abstraction can be added in System1 only.`

`b) An additional level of abstraction can be added in System2 only.`

`c) An additional level of abstraction can be added in both System1 and System2.`

`d) No additional level of abstraction can be added in either System1 or System2.`

Answer: b)
Explanation: In `System2` there is a common characteristic `Company` between `Delivery Service Agent` and `Paint Service Agent`, along with a common responsibility of `generates a new service request code`. Hence another layer of abstraction can be added in this system. Hence, option (b) is correct.

Question 3

In a library management system, a member is provided with an unique ID. The system also records the Books that have been checked out by the member, along with their due dates. Each member is also associated with a pending Fine Amount. A member can check out a book or return a book. If a book is returned after the due date, the member has to pay a fine.
Which of the following representations of a Member correctly corresponds this given scenario with respect to Abstraction and Encapsulation? *Marks: 2* **MCQ**

Abstraction: Member
Structure: *-ID* *-BooksCheckedOut* *-DueDates* *-PayFine*

Behavior:
+CheckOut()
+Return()
+CheckDueDate()

a)

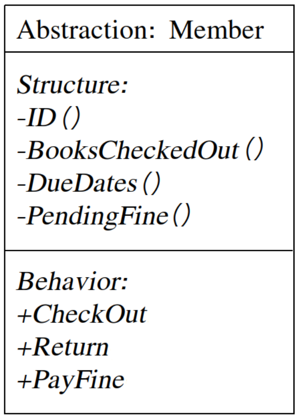
Abstraction: Member
Structure: *-ID* *-BooksCheckedOut* *-DueDates* *-PendingFine*

Behavior:
+CheckOut()
+Return()
+PayFine()

b)

Abstraction: Member
Structure: *-ID ()* *-BooksCheckedOut ()* *-DueDates ()* *-PayFine ()*
Behavior: *+CheckOut* *+Return* *+CheckDueDate*

c)



d)

Answer: b)
Explanation: Encapsulation will hide the implementation details. Hence, the variable names will not be public. The orifice or contractual interface through which the member interacts with the system will be public. Refer to Module 8, slide 25. Hence, option (b) is correct.

Question 4

Consider the following two code snippets:

PYTHON

```
1. vari=2.5
2. type(vari)
3. vari=20
4. type(vari)
```

C++

```
1. #include <iostream>
2. int main() {
3. float vari = 0.8;
4. std::cout << typeid(vari).name() << '\ n';
5. vari=2;
6. std::cout << typeid(vari).name() << '\ n';
7.}
```

`type(vari)` and `std::cout << typeid(vari).name() << '\ n'` prints the data type of `vari`. Which of the following is true? *Marks: 2* **MSQ**

a) `Line 2 in PYTHON prints the type of vari to be float but Line 4 in PYTHON prints the type of vari to be int.`

b) `Both Lines 2 and 4 in PYTHON print the type of vari to be float.`

c) `Line 4 in C++ prints the type of vari to be float (or f) but Line 6 in C++ prints the type of vari to be int (or i).`

d) `Both Lines 4 and 6 in C++ prints the type of vari to be float (or f).`

Answer: a), d)
Explanation: Options (a) and (d) are correct because PYTHON is dynamically typed and C++ is a statically typed language.

Question 5

In an Online Clothing System,the following functions are present:

1. `Select Quantity`
2. `Check Cashback`
3. `Cancel Product`
4. `Add to e-wallet`
5. `Add to bank account`
6. `Request Refund`
7. `Pay Amount`
8. `Confirm Selection`
9. `Request Replacement`
10. `Select New Product`
11. `Select Ordered Product`

There are 3 distinct modules of the system. If the first module includes function 10, second module includes function 2 and the third module includes function 11, which of the following options is true, based on the most optimal modular design? *Marks: 2* **MCQ**

```
a) Module 1 includes functions 5, 10, 8, 7.
   Module 2 includes functions 1, 2, 3, 4.
   Module 3 includes functions 11, 6, 9.
```

```
b) Module 1 includes functions 1, 10, 4, 5, 7.
   Module 2 includes functions 2, 8, 9.
   Module 3 includes functions 11, 3, 6.
```

```
c) Module 1 includes functions 1, 10, 8, 7.
   Module 2 includes functions 2, 4, 5.
   Module 3 includes functions 11, 3, 6, 9.
```

```
d) Module 1 includes functions 10, 7, 9.
   Module 2 includes functions 1, 2, 4, 5, 8.
   Module 3 includes functions 11, 3, 6.
```

Answer: c)
Explanation: From the given functions and the grouping of functions 10, 2 and 11, it can be estimated that the 3 most optimal modules should be `Ordering, Reward` and `Cancellation`. The grouping of the functions in option (c) suits these three modules in the most optimal way. Hence, option (c) is correct.

Question 6

Which of the following is a characteristic of a `Process`? *Marks: 2* **MCQ**

a) It is an alternative term for Lightweight concurrency.

b) Communication between `Processes` is expensive.

c) Two or more `Processes` may share the same address space.

d) Communication between `Processes` must involve shared data.

Answer: b)
Explanation: Refer to Module 10, slide 14. Hence, option (b) is correct.

Question 7

Which of the following is a Second-generation Language? *Marks: 2* **MCQ**

a) `ALGOL 58`

b) `Simula`

c) `ALGOL 68`

d) `Lisp`

Answer: d)
Explanation: Refer to Module 6, slide 10. Hence, option (d) is correct.

Question 8

Which of the following is (are) true about good modular designs? *Marks: 2* **MSQ**

a) They are comparatively easy to change.

b) They are comparatively difficult to understand.

c) Different modules heavily depend on each other.

d) They can be reused to develop other software systems.

Answer: a), d)
Explanation: Good modules are easy to change and easy to understand. They minimally depend on each other. They can also be reused.
Hence, options (a) and (d) are correct.

Question 9

A class hierarchy is designed through the following steps:

1. Class `Artist` is placed at the root of the hierarchy.
2. Classes `Dancer` and `Singer` inherit the properties of `Artist`.
3. Class `Street Performer` inherits the properties of both `Dancer` and `Singer`.
4. Class `Experienced Street Performer` inherits the properties of `Street Performer`.

Which of the above statements causes a problem of ambiguity of inherited properties in a sub-class?

Marks: 2 **MCQ**

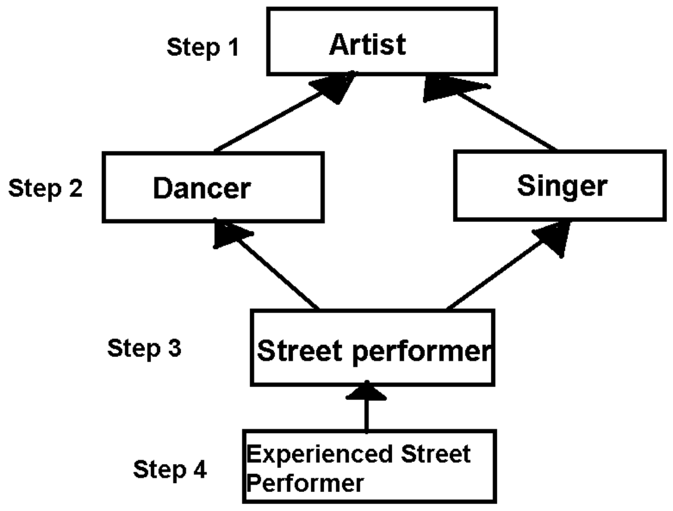
a) 1

b) 2

c) 3

d) 4

Answer: c)
Explanation: The problem of ambiguous inherited properties (diamond problem) occurs when the parents of a subclass has a common parent class. The hierarchy created by the given steps is as follows:



Step 3 creates the diamond problem. Hence, option (c) is correct.

Question 10

Consider the following scenario:
In an Online Video System, both `Subscribers` and content `Creators` are `Viewers`. A `Subscriber` can be an `Old` or a `New Subscriber`. A `Traveller` is a special type of `Creator` who can be either a `Local` or a `Global Traveller`.
Which of the following is true, if a class hierarchy of the Online Video System is to be constructed? *Marks: 2* **MCQ**

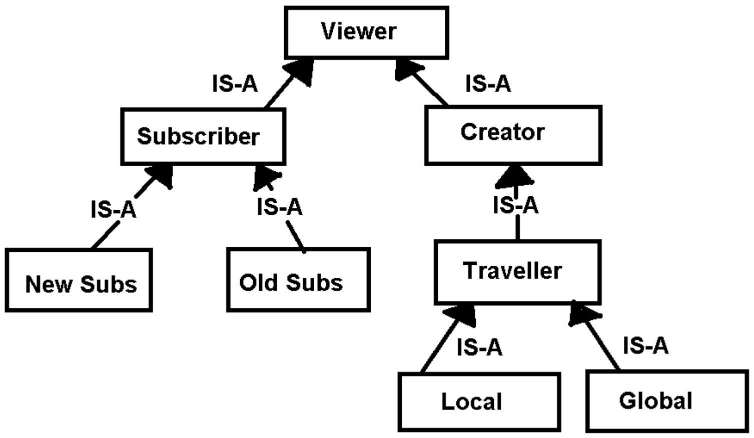
a) `Global Traveller` inherits the properties of `New Subscriber` through multiple inheritance.

b) `Traveller` inherits the properties of `Viewer` through multiple inheritance.

c) `Local Traveller` inherits the properties of `Viewer` through multi level (or hierarchical) inheritance.

d) `Old Subscriber` inherits the properties of `Global Traveller` through hybrid inheritance.

Answer: c)
Explanation: According to the given scenario, the following class hierarchy can be constructed:



Hence, option (c) is correct.