

QB365 Question Bank Software Study Materials

Data Abstraction Important 2, 3 & 5 Marks Questions With Answers (Book Back and Creative)

12th Standard

Computer Science

Total Marks : 75

2 Marks

10 x 2 = 20

1) What is abstract data type?

Answer : (i) Abstract Data type (ADT) is a type (or class) for objects whose behavior is defined by a set of value and a set of operations.

(ii) The definition of ADT only mentions what operations are to be performed but not how these operations will be implemented.

2) Differentiate constructors and selectors.

Answer :

S.No	Constructors	Selectors
(i)	Constructors are functions that build the abstract data type.	Selectors are functions that retrieve information from the data type.
(ii)	Constructors create an object, bundling together different pieces of information.	Selectors extract individual pieces of information from the object

3) What is a Pair? Give an example.

Answer : (i) Any way of bundling two values together into one can be considered as a Pair. Lists are a common method to do so. Therefore List can be called as Pairs.

(ii) **Example:** List = [10,20,30]

4) What is a List? Give an example.

Answer : List is constructed by placing expressions within square brackets separated by commas. List can store multiple values.

Example:

1st := [10, 20]

x, y:= 1st

In the above example x will become 10 and y will become 20.

5) What is a Tuple? Give an example.

Answer : (i) A tuple is a comma-separated sequence of values surrounded with parentheses. Tuple is similar to a list.

(ii) The difference between the two is that you cannot change the elements of a tuple once it is assigned whereas in a list, elements can be changed

(iii) Example : colour = ('red', 'blue', 'Green')

6) How the concrete level of data abstraction implemented?

Answer : (i) To implement the concrete level of data abstraction, languages like Python provides a compound structure called Pair which is made up of list or Tuple.

(ii) The first way to implement pairs is with the List construct

(ii) The first way to implement pairs is with the list construct

7) How the elements of a list can be accessed?

Answer : (i) The elements of a list can be accessed in two ways.
(ii) The first way is via multiple assignment and the second method is by the element selection operator

8) What is data abstraction?

Answer : Data abstraction is a powerful concept in computer science that allows programmers to treat code as objects.

9) Define Modularity. What will provide it?

Answer : Modularity means splitting a program into many modules. Abstraction provides modularity.

10) How many parts are there in a program? What are they?

Answer : There are two parts in a program. They are
(i) That part that operates on abstract data.
(ii) The part that defines a concrete representation.

3 Marks

10 x 3 = 30

11) Differentiate Concrete data type and abstract datatype.

Answer :

S.No	Concrete data type	Abstract data type
(i)	Concrete Data Types or structures (CDT's) are direct implementations of a relatively simple concept	Abstract Data Types (ADT's) offer a high level view (and use) of a concept independent of its implementation.
(ii)	A concrete data type is a data type whose representation is known.	Abstract data type the representation of a data type is unknown.

12) Which strategy is used for program designing? Define that Strategy.

Answer : A powerful strategy for designing programs, '**wishful thinking**'. Wishful Thinking is the formation of beliefs and making decisions according to what might be pleasing to imagine instead of by appealing to reality.

13) Identify Which of the following are constructors and selectors?

- (a) N1:=number()
- (b) accetnum (n1)
- (c) displaynum (n1)
- (d) eval (a/b)
- (e) x,y:= make slope (m), make slope (n)
- (f) display()

Answer : (a) Constructors
(b) Selectors
(c) Selectors
(d) Selectors
(e) Constructors
(f) Selectors

14) What are the different ways to access the elements of a list. Give example.

Answer : (i) The elements of a list can be accessed in two ways. The first way is via our familiar method of multiple assignment, which unpacks a list into its elements and binds each element to a different name.

```
lst := [10, 20]
```

```
x, y := lst
```

(ii) In the above example x will become 10 and y will become 20.

(iii) A second method for accessing the elements in a list is by the element selection operator. Unlike a list literal, a square brackets expression directly following another expression does not evaluate to a list value, but instead selects an element from the value of the preceding expression.

```
lst [0]
```

```
10
```

```
lst [1]
```

```
20
```

15) Identify Which of the following are List, Tuple and class ?

(a) arr [1, 2, 34]

(b) arr (1, 2, 34)

(c) student [rno, name, mark]

(d) day= ('sun', 'mon', 'tue', 'wed')

(e) x= [2, 5, 6.5, [5, 6], 8.2]

(f) employee [eno, ename, esal, eaddress]

Answer : (a) List

(b) Tuple

(c) Class

(d) Tuple

(e) List

(f) Class

16) Identify the constructor and selector from the following.

(i) City = Make city (name, lat, lon)

(ii) Get name (city)

(iii) Make point (x,y)

(iv) x coord (point)

(v) y coord (point)

Answer : (i) Constructor

(ii) Selector

(iii) Constructor

(iv) Selector

(v) Selector

17) Write a note on Data Abstraction.

Answer : (i) Data abstraction is supported by defining an abstract data type (ADT) which is a collection of constructors and selectors.

(ii) Constructors create an object, bundling together different pieces of information, while selectors extract individual pieces of information from the object.

18) How will you access the list elements using multiple assignments?

Answer : The elements of a list can be accessed in the method of multiple assignments, which unpacks a list into its elements and binds each element to a different name.

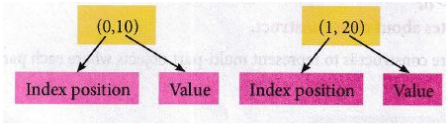
```
lst=[10,20]
```

```
x,y=lst
```

In the above example, x will become 10 and y will become 20. Mathematically we can represent list similar to a set.

Example:

```
lst [(0,10), (1,20)]
```



19) Write the pseudo code to represent the multi-part objects.

Answer : The structure construct is to represent multi-part objects where each part is named. Consider the pseudo code.

Class person:

```
creation()
```

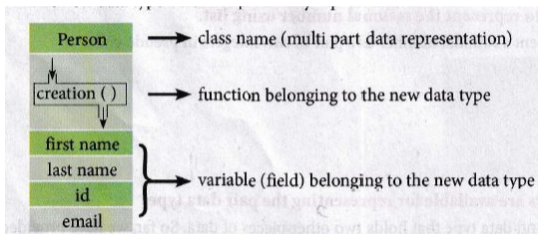
```
first name:= " "
```

```
last name:= " "
```

```
id := " "
```

```
email:= " "
```

The new data type Person is pictorially represented as



20) Compare List and Tuple:

Answer :

LIST	TUPLE
1. List uses square brackets	Tuple uses parentheses
2. Elements can be changed	Elements cannot be changed
3. Ex: lst:= [10,20]	Ex: nums:= (1,2)
4. Mutable sequence	Immutable sequence

5 Marks

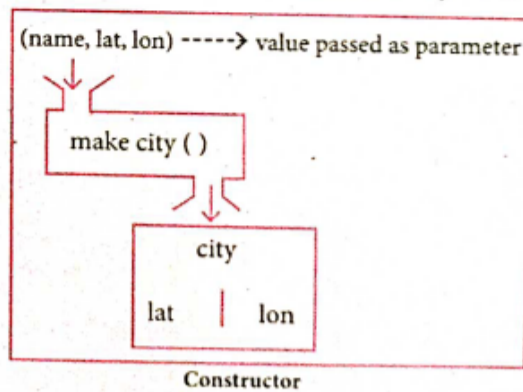
5 x 5 = 25

21) How will you facilitate data abstraction. Explain it with suitable example.

Answer : Data abstraction is used to define an Abstract Data Type (ADT), which is a collection of constructors and selectors. To facilitate data abstraction, you will need to create two types of functions : Constructors and Selectors.

Constructors :

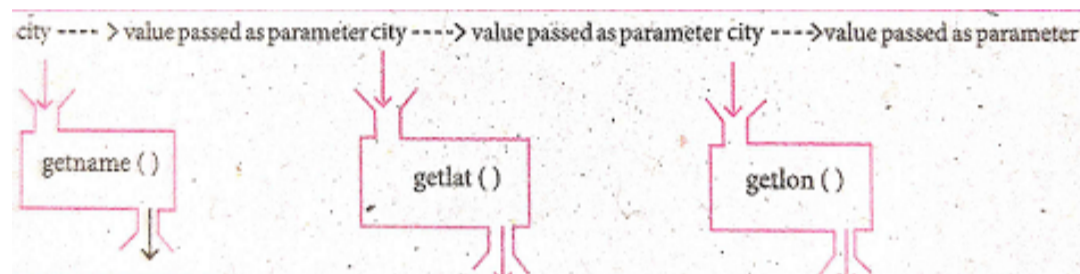
- (i) Constructors are functions that build the abstract data type.
- (ii) Constructors create an object, bundling together different pieces of information.
- (iii) For example, say you have an abstract data type called city.
- (iv) This city object will hold the city's name, and its latitude and longitude.
- (v) To create a city object, you'd use a function like city = makecity (name, lat, lon).
- (vi) Here makecity (name, lat, lon) is the constructor which creates the object city.



Selectors :

- (i) Selectors are functions that retrieve information from the data type.
- (ii) Selectors extract individual pieces of information from the object.
- (iii) To extract the information of a city object, you would use functions like
 getname(city)
 getlat(city)
 getlon(city)

These are the selectors because these functions extract the information of the city object.



22) What is a List? Why List can be called as Pairs. Explain with suitable example.

Answer : List :

(i) List is constructed by placing expressions within square brackets separated by commas. Such an expression is called a list literal. List can store multiple values. Each value can be of any type and can even be another list.

Example for List [10, 20].

(ii) The elements of a list can be accessed in two ways. The first way is via our familiar method of multiple assignment, which unpacks a list into its elements and binds each element to a different name.

lst := [10, 20]

x, y = lst

(iii) In the above example x will become 10 and y will become 20. A second method for accessing the elements in a list is by the element selection operator.

(iv) Unlike a list literal, a square-brackets expression directly following another expression does not evaluate to a list value, but instead selects an element from the value of the preceding expression.

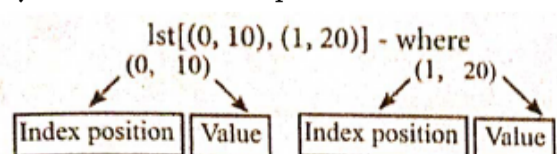
lst [0]

10

lst [1]

20

(v) In both the example mentioned above mathematically we can represent list similar to a set.



Pair :

Any way of bundling two values together into one can be considered as a pair. Lists are a common method to do so. Therefore List can be called as Pairs.

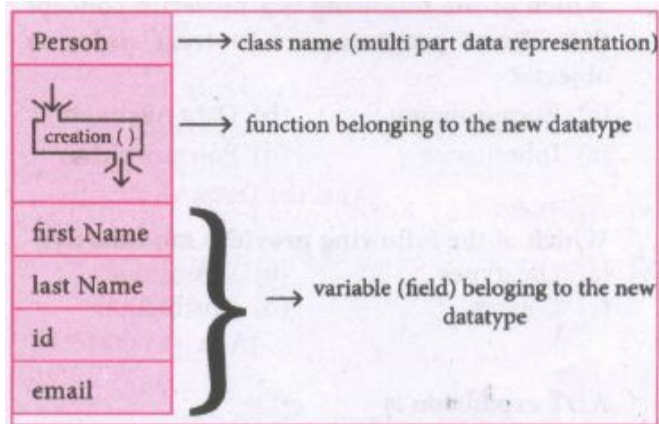
23) How will you access the multi-item. Explain with example.

Answer : (i) The structure construct (In OOP languages it's called class construct) is used to represent multi-part objects where each part is named (given a name). Consider the following pseudo code:

Class Person:

```
creation()
first Name:=""
last Name:=""
id :=""
email :=""
```

The new data type Person is pictorially represented as



In the case of something more complex, like a person, we have a multi-item object where each item is a named thing: the first name, the last name, the id, and the email. One could use a list to represent a person:

```
Person = ['padmashri', 'Baskar', '994-222-1234', 'compsci@gmail.com']
```

Let main() contains	
p1:=Person()	statement creates the object
first name:= "padmashri"	setting a field called firstName with value padamashri
last name:= "Baskar"	setting a field called last name with value Baskar
id:= "994-222-1234"	setting a field called id value 994-222-1234
email="compsci@gmail.com"	setting a field called email with value compsci@gmail.com
-- output of firstName: Padmashri	

(ii) The class (structure) construct defines the form for multi-part objects that represent a person.

(iii) Person is referred to as a class or a type, while p1 is referred to as an object or an instance.

(iv) Here class Person as a Cookie cutter, and p1 as a particular cookie, Using the cookie cutter you can make many cookies, Same way using class created many objects of that type.

(v) A class defines a data abstraction by grouping related data items. A class is not just data, it has functions defined within it, We say such functions are subordinate to the class because their job is to do things with the data of the class.

24) Explain the representation of Abstract datatype using rational numbers.

Answer : (i) The basic idea of data abstraction is to structure programs so that they operate on abstract data. That is, our programs should use data in such a way, as to make as few assumptions about the data as possible.

(ii) At the same time, a concrete data representation is defined as an independent part of the program.

(iii) Any program consist of two parts. The two parts of a program are, the part that operates on abstract data and the part that defines a concrete representation, is connected by a small set of functions that implement abstract data in terms of the concrete representation.

(iv) To illustrate this technique, let us consider an example to design a set of functions for manipulating rational numbers.

(v) **Example:** A rational number is a ratio of integers, and rational numbers constitute an important sub-class of real numbers.

A rational number such as 8/3 or 19/23 is typically written as : < numerator > / < denominator >

(vi) where both the < numerator > and < denominator > are placeholders for integer values. Both parts are needed to exactly characterize the value of the rational number. Actually dividing integers produces a float approximation, losing the exact precision of integers.

(vii) However, you can create an exact representation for rational numbers by combining together the numerator and denominator.

(viii) As we know from using functional abstractions, we can start programming productively before you have an implementation of some parts of our program.

(ix) Let us begin by assuming that you already have a way of constructing a rational number from a numerator and a denominator. You also assume that, given a rational number, you have a way of selecting its numerator and its denominator component.

25) Explain in detail about data abstraction?

Answer : (i) Data abstraction is a powerful concept in computer science that allows programmers to treat code as objects.

(ii) Programmers need not to worry about how code is implemented they have to just know what it does.

(iii) This is especially important when several people are doing a project. Project refers to the programming with data abstraction, our group members won't have to read through every line of our code to understand. They can just assume that it does work.

(iv) Abstraction provides modularity. It means splitting a program into many modules. Classes are the representation for "Abstract Data Types", (ADT).