Quiz 5 (November 26, 2020)

Total points 15/18

Write your name and ID correctly.
There are 18 questions in this quiz. Each question carries 1 mark. The total time duration for the quiz is 30 minutes. Answer the questions and submit your responses.

The respondent's email address (**f20181119@pilani.bits-pilani.ac.in**) was recorded on submission of this form.

0 of 0 points

ID *

2018A7PS1119P

Name *

Shreyas Bhat Kera

Questions 1-18 15 of 18 points

Which keyword is used in C++ for dynamic binding of messages to method? 1/1

virtual

Which statements are correct in reference to the functions in Haskell programming language (1) language supports polymorphism (2) functions are always recursive (3) functions can be non-recursive (4) functions always use the accumulator (5) language does not support higher order functions 1/1

- ○ None of these
- ◉ Statements 1 and 3
- ○ Statements 1, 2 and 5
- ○ Statements 1, 3 and 5
- ○ Statements 2, 4 and 5
- ○ Statements 2 and 4
- ○ Statements 1, 3, 4 and 5

If the type signature of a function (say f) in Haskell programming language is Int->Int->Int->Bool, then if bracketing is used to show the associativity, this would internally mean 1/1

- ◉ Int->(Int->(Int->Bool))
- ○ (Int->Int->Int)->Bool
- ○ None of these
- ○ ((Int->Int)->Int)->Bool

Following statements hold good with respect to the purest model of object oriented computation with exclusive use of objects and without any mix of imperative language types. (1) Simple operations are done through message passing. (2) This model works faster than the one with added imperative language types. (3) All types are classes 1/1

- ○ Statements 1, 2 and 3
- ◉ Statements 1 and 3
- ○ None of these
- ○ Statements 1 and 2
- ○ Statements 2 and 3

Which of the following statements are correct with respect to the derived class being called as a subtype. (1) A derived class is a subtype if it has an "is-a" relationship with its parent class. (2) The number of parameters of the overriding method and that of the overridden method are same. (3) The type of parameters of the overriding method and that of overridden method are different (4) public entries of the parent class are inherited as public entities in the subclass. (5) A subtype cannot have methods different from the parent class

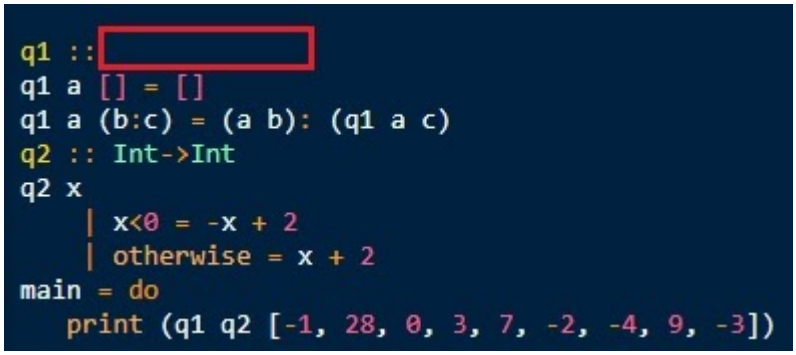
0/1

- ○ Statements 1, 2, 4 and 5 are correct
- ○ Statements 1, 2 and 4 are correct
- ◉ Statements 1, 3 and 4 are correct
- ○ Statements 2, 3 and 4 are correct
- ○ Statement 2, 4 and 5 are correct
- ○ None of these

Correct answer

- ◉ Statements 1, 2 and 4 are correct

Consider the following function written in Haskell programming language. Which piece of code should be written inside the given box to get the output as [3,30,2,5,9,4,6,11,5] ? [Ensure that the program becomes syntactically and semantically correct after filling your answer] 1/1



- (Int->Int)
- (Int->Int)->[Int]
- None of these
- [Int]
- [Int]->[Int]
- Int->Int
- (Int->Int)->[Int]->[Int]

Object slicing refers to 1/1

- user defined selection of information to be copied into variable of parent type if the objects are heap dynamic
- None of these
- loss of information while assigning the data of subtype to a variable of parent class if the objects are stack dynamic.
- user defined selection of information to be copied into variable of parent type if the objects are stack dynamic
- loss of information while assigning the data of subtype to a variable of parent class if the objects are heap dynamic.

Which of the following statements are correct with respect to C++ Object Oriented Programming language? (1) The stack dynamic object's type is known at compile time (2) The heap dynamic object type is only known at run time (3) The message to method binding is always done at compile time (4) Overiding methods and overridden methods have exactly same parameter profiles (5) The messages to method binding is always dynamic(6)The stack dynamic variables are polymorphic. 1/1

- Statements 1, 2 and 4
- Statements 2, 3 and 4
- Statements 4, 5 and 6
- None of these

State whether the statement "Private-derived subclasses cannot be subtypes" is true or false. 1/1

- ○ False
- ◉ True

Consider the following code written in Haskell language whose output is [3,0,5,2]. What is the type of the function f1? [Ensure that the program becomes syntactically and semantically correct after filling your answer] 1/1

```
f1 ::
f1 [] = []
f1 (x:y) = len(x): f1 y
  where len :: [Int]->Int
        len [] = 0
        len (u:y) = 1 + len y

main = do
    print(f1 [[2,4,1], [], [90, 8, 78, 67, 95], [19, 20]])
```

[[Int]]->[Int]

Feedback

small case letter i in type specification in place of upper case I is considered wrong.

The purpose of the constructor method in Object Oriented Programming is to 0/1

- ○ None of these
- ○ allocate memory if the data member is the pointer to heap dynamic data but it does not initialize the data members
- ○ allocate memory if the data member is the pointer to heap dynamic data and initialize the data members
- ◉ allocate memory if the data member is stack dynamic data and initialize the data members

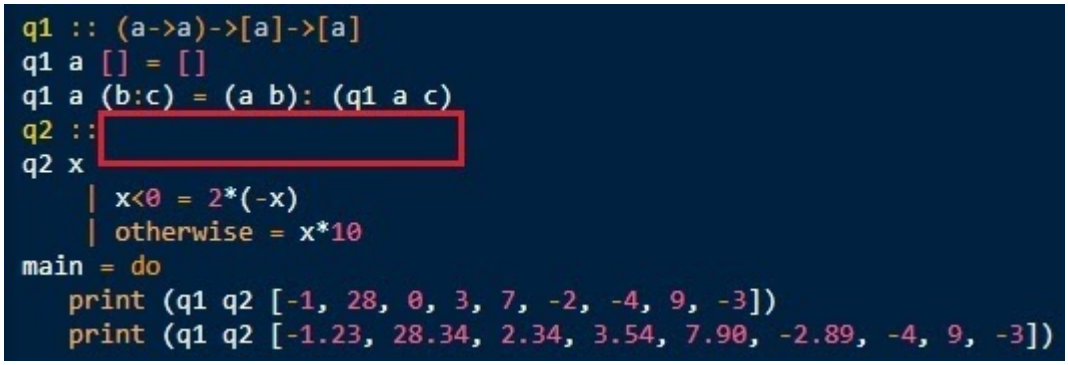
Correct answer

- ◉ allocate memory if the data member is the pointer to heap dynamic data and initialize the data members

Which of the following Languages follows pure object oriented paradigm? 1/1

- ○ Python
- ○ Java
- ○ None of these
- ○ C
- ◉ Smalltalk

Consider the following code written in Haskell programming language. The functions q1 and q2 work well with integers and real numbers. The polymorphic type a (small letter) is used in the type descriptions. The function q2 deals with integers at one time and can also deal with real numbers at the other time. What should be written in the rectangular box in order to be able to get the output of the program as [2,280,0,30,70,4,8,90,6] [2.46,283.4,23.4,35.4,79.0,5.78,8.0,90.0,6.0]? [Ensure that the program becomes syntactically and semantically correct after filling your answer] 0/1



```
q1 :: (a->a)->[a]->[a]
q1 a [] = []
q1 a (b:c) = (a b): (q1 a c)
q2 ::
q2 x
    | x<0 = 2*(-x)
    | otherwise = x*10
main = do
   print (q1 q2 [-1, 28, 0, 3, 7, -2, -4, 9, -3])
   print (q1 q2 [-1.23, 28.34, 2.34, 3.54, 7.90, -2.89, -4, 9, -3])
```

a->a

Correct answers

(Ord a, Num a)=> a->a

(Num a, Ord a) => a -> a

If class A and class B are derived from a common parent Z, and a class C has both A and B as parent classes, then this situation is called as 1/1

- None of these
- Polymorphism
- Diamond inheritance
- Object slicing
- Subtype

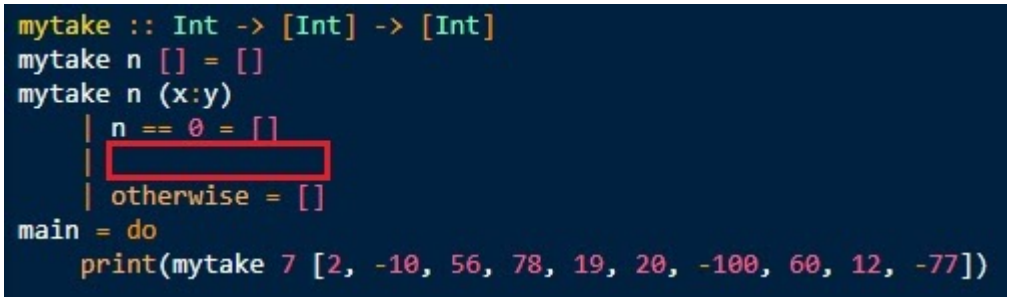
The technique of rewriting or implementing a function of multiple arguments into a sequence of functions with a single argument in a functional programming language such as Haskell is called 1/1

- ○ None of these
- ◉ Currying
- ○ Overriding
- ○ Object slicing
- ○ Polymorphism

If the objects in an Object Oriented language are allocated space on the heap, and the deallocation is explicit, then the problem comes in 1/1

- ◉ handling of the dangling pointers
- ○ handling of the memory leaks
- ○ None of these
- ○ the reclamation of the memory

The Haskell programming uses an inbuilt function take applied on two arguments say n and L where n is the number of elements and L is the list. The function returns a list with first n elements of L. Consider a user version of take as mytake given as below and write the appropriate code to fill up the box so as to get the output of the program as [2,-10,56,78,19,20,-100]. [Ensure that the program becomes syntactically and semantically correct after filling your answer] 1/1



n>0 = x:mytake (n-1) y

Correct answers

n>0 = x: (mytake (n-1) y)

n>0 = x : mytake (n-1) y

n > 0 = x : mytake (n-1) y

n>0 = x: mytake (n-1) y

n > 0 = (x : (mytake (n-1) y))

n > 0 = x : (mytake (n-1) y)

n>0 = x: mytake (n-1) (y)

n>0 = x : (mytake (n-1) y)

n > 0 = x : mytake (n - 1) y

Correct Answer

Feedback

All answers with variations in n>0 = x: mytake (n-1) y are considered correct. Variations considered are the presence or absence of blank spaces and parentheses pair around y as (y) or around mytake (n-1) y as (mytake (n-1) y). If n-1 is without parentheses pair, answer is considered wrong. Answers like n>0 = x : mytake n-1 y are wrong.

Consider the following code written in Haskell programming language. Which piece of code should be written inside the given box to get the output as 15? [Ensure that the program becomes syntactically and semantically correct after filling your answer] 1/1

```
f1 :: Integer -> Integer
f1 n
    | n <= 0 = 1
    |otherwise = n * f1 [          ]
main = do
   print (f1 5)
```

(n-2)