

STD XII

Computer Science

Chapter 1 - Function

Two Mark Questions

Book Back Questions

1. What is a subroutine?
2. Define function with respect to programming language
3. Write the interference you get from $X := (78)$
4. Differentiate interface and implementation
5. Which of the following is a normal function definition and which is recursive definition
 - a)

```
let rec sum x y:  
    return x + y
```
 - b)

```
let disp :  
    print 'welcome'
```
 - c)

```
let rec sum num:  
    if(num!=0) then return num + sum (num-1)  
    else  
    return num
```

Extra Questions

1. Define parameters and arguments
2. For which set of statements subroutines are useful?
3. What is an external interface?
4. Define recursive function
5. State the purpose of an interface
6. What does the code $x1 \rightarrow \dots \rightarrow xn \rightarrow y$ denote ?
7. Which criteria is necessary in comparison of algorithms
8. What are the inputs of a function code?
9. Give an example for an interface in real world situation
10. Why is a function called as impure function

Three Mark Questions

Book Back Questions

1. Mention the characteristics of an interface
2. Why strlen is called pure function?
3. What is the side effect of impure function. Give example
4. Differentiate pure and impure function
5. What happens if you modify a variable outside the function? Give example

Extra Questions

1. What does require and return options of a function indicate?
2. What does let statement denote?
3. In object oriented programming language what is the role of an interface?
4. How is external interface and an object related?
5. Why is impure function considered to be good?

6. What are the advantages of having pure functions?
7. What happens when you modify the variable inside a function?
8. Write a recursive function code to check whether the given number is odd or even
9. Draw the flow chart for the given algorithm

monochrome (a, b, c)

- - inputs : a = A, b = B, c = C
- - outputs : a=b=0, c = A+B+C

While a > 0

a ,b ,c := a-1 b-1 c+2

Five Mark questions

Book Back Questions

1. What are called parameters. Write a note on parameters with type and parameters without type
2. Explain with example pure and impure functions
3. Explain with an example interface and implementation
4. Identify in the following program
let rec gcd a b :=
if b <> 0 then gcd b(a mod b) else return a
 - a) Name of the function
 - b) Identify the statement which tells it is a recursive function
 - c) Name of the argument variable
 - d) Statement which invokes the function recursively
 - e) Statement which terminates recursion

Extra questions

1. What is an interface? What are the characteristics of an interface. Explain with an example
2. What are the side effects of impure function? Explain with an example the outcome of modifying the variable outside a function.
3. Write an algorithm to find the greatest common factor of two positive integers using pure functions and explain the same.
4. With respect to programming language are subroutines and functions one and the same – justify your answer
5. Construct an algorithm that arranges meeting between two types of chameleon that changes the color to the third type.

STD XII

Computer Science

Chapter 2 – Data Abstraction

Two Mark Questions

Book Back Questions

1. What is a abstract data type?
2. Differentiate constructors and selectors
3. What is a pair? Give an example
4. What is a list? Give an example
5. What is a tuple? Given an example

Extra Questions

1. What does data abstraction provide?
2. How is data abstraction facilitated?
3. Mention the constructors and selectors in the following code
City = makecity (name lat lon)
Getlon(city)
Getname(city)
4. What is the necessity of abstraction?
5. Define concrete data type
6. Name the two parts of a programme?
7. Define a class
8. Differentiate a class and an object
9. What does multi item object mean?
10. What does the statement p1= person () denote?

Three Mark Questions

Book Back Questions

1. Differentiate concrete data type and abstract data type
2. Why which strategy is used for program designing? Define that strategy
3. Identify the constructors and selectors from the following code
 - a) N1 = number()
 - b) Acceptnum(n1)
 - c) Displaynum(n1)
 - d) Eval(a/b)
 - e) x,y = makeslope(m), makeslope(n)
 - f) display ()
4. What are the different ways to access the elements of a list. Give example
5. Identify which of the following are List, Tuple and Class
 - a) Arr [1,2,3,4]
 - b) Arr(1,2,3,4)
 - c) Student[rno,name,mark]
 - d) Day = ('sun', 'mon', 'tue', 'wed')
 - e) X = [2,5,6,[5,6],8,2]
 - f) Employee[eno, ename, esal, eaddress]

Extra Questions

1. What does require and return options of a function indicate?
2. What does let statement denote?
3. In object oriented programming language what is the role of an interface?
4. How is external interface and an object related?
5. Why is impure function considered to be good?
6. What are the advantages of having pure functions?
7. What happens when you modify the variable inside a function?
8. Write a recursive function code to check whether the given number is odd or even
9. Draw the flow chart for the given algorithm

monochrome (a, b, c)

- inputs : a = A, b = B, c = C
- outputs : a=b=0, c = A+B+C

While a > 0

a, b, c := a-1 b-1 c+2

Five Mark questions

Book Back Questions

1. What are called parameters. Write a note on parameters with type and parameters without type
2. Explain with example pure and impure functions
3. Explain with an example interface and implementation
4. Identify in the following program
let rec gcd a b :=
if b <> 0 then gcd b(a mod b) else return a
 - a) Name of the function
 - b) Identify the statement which tells it is a recursive function
 - c) Name of the argument variable
 - d) Statement which invokes the function recursively
 - e) Statement which terminates recursion

Extra questions

1. What is an interface? What are the characteristics of an interface. Explain with an example
2. What are the side effects of impure function? Explain with an example the outcome of modifying the variable outside a function.
3. Write an algorithm to find the greatest common factor of two positive integers using pure functions and explain the same.
4. With respect to programming language are subroutines and functions one and the same – justify your answer
5. Construct an algorithm that arranges meeting between two types of chameleon that changes the color to the third type.