Assignment 1 Data Structures & Algorithms. (PRD190001)

OA I Prove by Conthi

Q1] Suppose your calculator only did base 10 algorithms. write an expression to compute log base 2 of 2048 using only log base 10.

$$109_{2}2048 = \log_{10} 2048$$

expression that can be directly computed from k)

3+5+7+9+...+2k+1

$$= K(K+1)+K$$

Prove by counter example. It is false.

Prove the following statement is false. $n^3 > 2^n$ for any n > 1

$$\rightarrow 10^3 = 1000 : 10^3 \neq 2^{10}$$

$$2^{10} = 1024$$
 . The above statement is false.

(19 00 plan) smilling plan 2 22 pulsual plan 1 trismangles A.

Paove by contradiction.

Prove the following statement is true.

The square of even numbers is also even?

> Let us assume that square of even number is odd.

Thus by squaring the even number 2n we get $2n \times 2n = 4n^2$.

4n2 \$4 4n2 +1 ommus primolo 3 + sarqx3 [20

Also 4n2 is an even number.

Hence our assumption was incorrect the square of even numbers is always even.

Q5a) Prove by induction "+ (s)(1+mm)s=

$$\sum_{i=1}^{n} i^3 = [n^2][(n+1)^2]/4x + (2)(1+x)x)e =$$

Base case :-

$$\frac{1}{1} \int_{1}^{2} 1^{3} = 1^{2} \left[(1+1)^{2} \right] \sqrt{1 + (1+1)^{2}}$$
where the following statement is also after the following statement is also as the foll

This holds true.

The above statement is

210 = 1024

Us > 50 to and US T

> 103 = 1000 108 \$ 210

Let us assume it holds true for k (until k) Of write Joya or pseudos $\sum_{i=1}^{k} i^3 = (k^2) \left[(k+1)^2 \right] / 4$ a value displays ? for kit 1 - m) (8 - m) (8 - m) (8 - m) (2 - m) (6 - m) n $\stackrel{k+1}{\leq} = i^{3} = (k^{2}) \left[(k+1)^{2} \right] / 4 + (k+1)^{3}$ FUNCTION PRINTURDER (17) & = [k2][(k+1)2]+4(k+1)3 PRINTORDER (N-1) 4 $= \left[k^2 + 4k + 4 \right] \left[k + 1 \right]^2$ 30==n 71 3210 4 i we have proved that 10 29 13 = [n2] [(n+1)2]/4 holds toue. array of integers and a position as politichers and octuans the count of odd numbers in angly-Q5b) n²-n is even for any n>1 Base case: for n= 1 12-1=0 is true. Sprig +xon no result Let us assume that this holds true until k 1. K2-K= Oven trye. ! for (k+1) $(k+1)^{2}-(k+1)=k+2k+1-k-1$ $=2k \leftarrow even$ in n2-n is even for any n=1

```
et us assume it holds true for k (until k)
Q6] write Java or pseudocode:
 a) write a recursive function when passed
a value displays
  n(n-1)(n-2)(n-3)...o...(n-3)(n-2)(n-1)n
          K = 13=(K2)[(K+1)2]/4+(K+1)
 FUNCTION PRINTORDER (n) {
      if n>0 {
  print(n) + [ (+x)][ x]
  PRINTORDER (n-1)
       y print(n) = [1+x][++x++=x]
   else if n == 0 {
  print (0)
 is we mave proved that I saydwan grang to
Q6b] write a recursive function that receives an
```

array of integers and a position as parameters and returns the count of odd numbers in augy. Let each recursive call consider next integer in Base case: for n= 1 assay,

-> Answer on next page. sunt 21 0=1=51 Let us assume that this holds have until lo

K2-K - Over hule. (1+x) rot ! (k+1)2-(k+1)=k+2k+1-k-1 = 2x <- even : N2-11 is even for any n21

```
-> FUNCTION COUNTODD (arr, pos) &
if pos > arr.length - 1 {
         print (incorrect position)
                         P. set Object 1 ( 60 hello
      else if pos <0 { or brown 3) stopped
          print (in correct position)
      else if pos == arr, length = 1 &
          if arr[pos] 1, 2 == 0 ?
               return 0 4
           else return 1
       else {
          if arr [pos] 1, 2 == 0
             return COUNTODD (arr, pos+1)
          else
             return 1+ countodd (arr, pos + 1)
```

Q7) Suppose there exists a generic Java class named Pair with type parameter T that stores two objects with get I set method for each. Write the statements necessary to create an object of type pair with string as its type parameter, I use the set method to set the two strings, then the get methods to retreive it for printing. Note: You don't have to write the pair claus itself.

-> Pair < String, String > p = new Pair < String, String);

P. set Object 1 (66 hello 99);

P. set Object 2 (66 world 99);

String p1 = p. get Object 1 ();

String p2 = p. get Object 2 ();

String p2 = p. get Object 2 ();

39219

return Countopp (am, pos+4)

61SE

return 1+ countodd (arr, pos + 1)

(1) Suppose there exists a generic Java class named Pair with type parameter I that stones two objects with get & set method for each write the statements necessary to create an object or type pair with string as its type parameters. I use the set method to set the two strings, then the get methods to attack to for printing, water the door that it for printing, water that door that the pair class itself.