ASSIGNMENT – 9

Q1)

Lets suppose :- "f" be a PARTIALLY COMPUTABLE FUNCTION.
There exists a L-program that computes the function, f.
Adding a no-op instruction to L-program, also computes f by new L-program.
∴ By adding no-op instructions to the L-program, we can create new L-program that computes f.

Lets suppose :- We are adding "K" no-op instructions to generate new L-program where "K" can be any natural number [1,2,3 - - - - ∞]
∴ It can generate INFINITE L-programs and the length of the programs is
$\boxed{l \geq K,\ K \geq 1}$

Therefore, f is computed by INFINITELY many L-programs whose length
$\boxed{l \geq K,\ K \geq 0}$

Q2) Proof by Induction :-

→ BASE CASE :-

For $K=0$; the length of $P=0$

Therefore :- P is an empty program.

For empty program :- $\psi_P^{(1)}(x)=0$

This satisfies the condition :- $\psi_P^{(1)}(x)\leq 0$ as $0\leq 0$

→ Assume that, P is a straight line L-program of length m, then

$\psi_P^{(1)}(x)\leq n \longrightarrow (i)$

We are supposed to prove for the program of length $n+1$.

The program of length $n+1$ can be written as a :-

PROGRAM of length n and one extra instruction at the end of it.

From (i):-

The maximum value of the output of the program of length "n" is x

Adding one extra instruction, the maximum value can be $n+1$.

∴ the maximum value of the output of the program of length $n+1$ is $n+1$.

Therefore :- $\psi_P^{(1)}(x)\leq n+1$, if the length is $n+1$.

Hence, by PROOF OF INDUCTION, if length of program P is K,

then $\psi_P^{(1)}(x)\leq K$ for all x.

Q3) To prove that a function is p.c in L++ $\Leftrightarrow$

It is a p.c 2 conditions must be satisfied.

(i) A function can be computed by L++ program if it can be computed by L-program

(ii) A function can be computed by L program if it can be computed by L++ program

As all the instruction in L are included in L++, we can obviously say that a function computed by L program can also be computed by L++ program.

→ By this, condition (i) is satisfied as we know that L++ extends L and all instructions in L-program can be executed by L++.

→ For condition (ii) to be satisfied, we should find a way for $V \leftarrow K$ to be computed in L-program.

The following code can be used for event $V \leftarrow K$ in L-program.

```
[A]   IF  V==0  GOTO B
      V ← V-1
      GOTO A
[B]   V → V+1  ⎫
      V → V+1  ⎪
      V → V+1  ⎬ ≡ [K-times]
         ⋮     ⎪
      V → V+1  ⎭
```

Since, L++ consists of all L-instruction and $V \leftarrow K$ can be computed by L program in the above method, we can say that condition (ii) is met.

Since, both conditions are satisfied, a function is p.c in L++ $\Leftrightarrow$ it is p.c