**TASK NO.1:**

A newly developed compiler for C was found to be much slower than expected .After some preliminary investigation ,it was suspected that the compiler was spending too much time manipulating the symbol table.

Show how the fraction of time used for symbol table manipulations, denoted *frac,* can be measured accurately using *software monitoring.*

**Solution:**

Time spent manipulating symbol table depends on program being compiled need set of randomly selected programs ,or specifically selected set designed to be representative of actual workload .use statistical techniques to analyze data

Tot\_time=time needed for compilation

ST\_time=time spent manipulation symbol table

Frac=ST-time/Tot\_time

**Computing Tot\_time:**

1-At he beginning of compilation initialized virtual clock

2-At end,read clock

3-can modify compiler code or by creating command line code(batch file) or calling program that does not clock ops and calls compiler

**Computing ST\_time:**

Modify compiler

1-Add fla variable

2-identify procedures that comprise ST\_handling function

3-add measurement statements that depend on flag

**Compiler Routine:**

Procedure pn(parameter list){

Int start,end;

If(measure\_flag)

Start=read\_virtual\_cliock()

………….

If(measure\_flag){

Finish=read\_virtual\_clock();

ST\_time+=(start-finish)

}

}

**TASK NO.2:**

Consider the measurement of the relative frequency of instruction usage.

a)Discuss pure hardware monitoring and software monitoring solutions for the given problem.

b)Discuss the drawbacks (if any) of both of these methods.

**Solutions(Part A):**

**Hardware Approach:**

1-Pick instruction opcode from data bus when instruction is fetched

2-user countr array,initialized to zeros

3-increment count[f(opcode)]

4-Relative freq=count[n]/sum of counts

**Software Approach:**

1-Counter array in main memory, initialized to zero

2-“Trap” after every instruction

* Can be done on some machines
* Turn off trapping
* Increment counter[opcode]
* Turn on trapping
* Return

**Solution(Part B):**

**Drawbacks of Hardware Approach:**

Following are the some important disadvantage of hardware approach including;

1-requires substantial HW

2-inflexible

**Drawbacks of Software Approach:**

Following are the some important advantage of software approach including;

1-Huge overhead

2-trap on every instruction