Homework # 1

1. Explain each of the following terms in your own words:

- **a) Translator.**
- **b) Interpreter.**
- **c) Virtual machine.**

Solution:

a. A translator converts programs in one language to another.

b. An interpreter carries out a program instruction by instruction.

c. A virtual machine is a conceptual machine, one that does not exist.

2. Is it conceivable for a compiler to generate output for the micro architecture level instead of for the ISA level? Discuss the pros and cons of this proposal.

Solution: It is possible, but there are problems. One difficulty is the large amount of code produced. Since one ISA instruction does the work of many microinstructions, the resulting program will be much bigger. Another problem is that the compiler will have to deal with a more primitive output language, hence it, itself, will become more complex. Also, on many machines, the microprogram is in ROM. Making it user-changeable would require putting it in RAM, which is much slower than ROM. On the positive side, the resulting program might well be much faster, since the overhead of one level of interpretation would be eliminated.

3. Can you imagine any multilevel computer in which the device level and digital logic levels were not the lowest level? Explain.

Solution: During the detailed design of a new computer, the device and digital logic levels of the new machine may well be simulated on an old machine, which puts them around level 5 or 6.

4. Consider a multilevel computer in which all the levels are different. Each level has instructions that are m times as powerful as those of the level below it, that is, one level r instruction can do the work of m level r-1 instructions. If a level 1 program requires k seconds to run, how long would equivalent programs take at levels 2, 3 and 4 assuming n level r instructions are required to interpret a single r+1 instruction?

Solution: Each level of interpretation slows down the machine by a factor of n/m.
Thus the execution times for levels 2, 3, and 4 are kn/m, kn^2/m^2, and kn^3/m^3, respectively.

5. Some instructions at the operating system machine level are identical to ISA language instructions. These instructions are carried out directly by the micro program rather than

by the operating system. In light of your answer to the preceding problem, why do you think this is the case?

Solution: Each additional level of interpretation costs something in time. If it is not needed, it should be avoided.

6. Consider a computer with identical interpreters at levels 1, 2, and 3. It takes an interpreter n instructions to fetch, examine and execute one instruction. A level 1 instruction takes k nanoseconds to execute. How long does it take for an instructions at levels 2, 3 and 4?

Solution: You lose a factor of *n* at each level, so instruction execution times at levels 2, 3, and 4 are *kn*, *kn^2*, and *kn^3*, respectively.

7. In what sense are hardware and software equivalent? Not equivalent?

Solution: Hardware and software are functionally equivalent. Any function done by one can, in principle, be done by the other. They are not equivalent in the sense that to make the machine really run, the bottom level must be hardware, not software. They also differ in performance.

8. Babbage's difference engine had a fixed program that could not be changed. Is this essentially the same thing as a modern CD ROM that cannot be changed? Explain your answer.

Solution: Not at all. If you wanted to change the program the difference engine ran, you had to throw the whole computer out and build a new one. A modern computer does not have to be replaced because you want to change the program. It can read many programs from many CD-ROMs.

10. The performance ratio of the 360 model 75 was 50 times that of the 360 model 30,yet the cycle time was only five times as fast. How do you account for this discrepancy ?

Solution: There are several reasons behind this. Raw cycle time is not the only factor. The number of bytes fetched per cycle is also a major factor, which is increasing with the larger models. There are other factors which also play important roles. These are ,memory speed and wait states as well as the presence of caching. Also, a better I/O architecture causes fewer cycles to be stolen, and so on.

12. Suppose that each of the 300 million people in the US fully consumes two packages of goods a day bearing RFID tags. How many RFID tags have to be produced annually to meet that demand? At a penny a tag, what is the total cost of the tags? Given the size of GDP, is this amount of money going to be an obstacle to their use on every package offered for sale?

Solution: Each person consumes 730 tags per nonleap year. Multiply by 300 million and you get 219 billion tags a year. At a penny a tag, they cost $2.19 billion dollars a year. With GDP exceeding $10 trillion, the tags add up to 0.02% of GDP, not a huge obstacle.