y2k?

Chandrama Shaw*

INTRODUCTION :

The 'year 2000' is the computer problem. This problem is also known as 'y2k problem'. The problem directly or indirectly affect nearly everyone on or after January 1, 2000. After the end of that century the computer cann't distinguish between year 1900 and 2000, between 1901 and 2001, 1902 and 2002 etc. The computer will perform the malfunction.

Why is this going to happen ?

Years ago, when memory and storage were small and costly, for efficiency reasons most computer programs used only two bytes** (storage spaces) to indicate the year.

For example : The year 1967 was simply written as

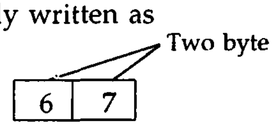

Fig-1

The computer understood that to mean 1967, when the clock strike 12 : 01 on January 1, 2000, the year will be shown as

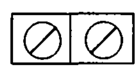

Fig-2

The computer then understood to be January 1, 1900.

PROBLEMS : The following three problems are commonly associated with the y2k problem.

- The roll over problem.
- The two-digit problem
- The leap-year problem.

● **The roll over**—each PC contains two time keeping devices, which might or might not recognise that the digits '00' indicate the year 2000. There are CMOS real time clock (RTC) also called the system clock and BIOS.

CMOS RTC, which keeps time whether the PC is on or off, notes the time, the date, the last two digits of the year. When the PC is 'on', the booting is started, then the BIOS retrieves the day of the month, month and the year from CMOS RTC and then the century.

BIOS simply counts the number of records that pass. It does not access CMOS RTC again until the PC is rebooted.

The CMOS RTC in each PC will roll over smoothly from 99 to 00. The BIOS will also seem to roll over smoothly because it simply counting sounds and the difference between December 31, 1999 23 : 59 : 59 and Jan 1, 2000 ; 00 : 00 : 00 is on tick.

In PCs thus are y2k ready, BIOS will record the change in century to CMOS's address, where

* Final Year Student of Computer Sc & Tech.

** Byte : 1 byte consists of 8 bits.

the two digits representing the base century.

- **Two digit problem :**

The two digit problem is actually an application based problem instead of hardware. Many application base date calculations on a two-digit year either ignored the century entirely or assuming that the two-digit year should always be preceded by 19. When we consider that the year 2000 ends with two zeros, it does not take long to realise that date calculation, which are fundamental to virtually every business computing process are not going to work properly.

For example, in the year 2000, applications that use only a two-digit in date calculations might determine that a person born in 1980 is minus eighty years old (00–80 = – 80), rather than twenty years old (2000–1980 = 20).

- **The Leap year problem :**

In addition to roll over problem and two-digit problem, the leap year problem occurs because the year 1900 was not a leap year while the year 2000 is. As a result, hardware and software that interpret the digits 00 as year 1900 will not handle properly. Even hardware and software that support a four-digit year or interpret the digit 00 as the year 2000 might not recognise the year 2000 as a leap year. Hardware that does not recognise the year 2000 as a leap year will be unable to accept February 29, 2000 as a correct date.

This will mishandle the following calculations :

* **Day-of-the-week calculations :**

February 28, 2000 is a Monday, and March 1, 2000 is Wednesday, not a Tuesday.

* **Day-of-the-year calculations :**

The year 2000 has 366 days, not 365 days.

* **Week-of-the-year calculations :**

The eleventh week of the year 2000 begins on March 13, not March 14.

This causes the miscalculation of interest in the Bank specially.