Саратовский государственный университет им. Н. Г. Чернышевского

**Е.В. Тиден**

# INFORMATION TECHNOLOGY

**Учебно-методическое пособие по английскому языку для студентов, обучающихся на механико-математическом факультете и факультете КНиИТ**

2015

**Тиден Е.В.** Информационные технологии: учебное пособие по английскому языку для студентов, обучающихся на механико-математическом факультете и факультете КНиИТ – Саратов, 2015 - 69 с.

Пособие представляет собой сборник текстов для чтения по специальности компьютерные технологии. В качестве проверки понимания прочитанного и для расширения словарного запаса предлагаются упражнения.

Для студентов старших курсов механико-математического факультета и факультета КНиИТ.


*Рецензент:*


Кандидат филологических наук, доцент, Н.Ю. Смирнова

# Содержание

# INTRODUCTION

The book is intended for students of intermediate level and above.

The texts included in the book are provided with terminology used by the scientists working in the area of Information Technology.

The book is written for 2$^{nd}$ & 3$^{rd}$ year students of Mathematical & Information Technology Departments. The book makes computer terminology easier to use and learn. It contains more than 300 terms.

The book consists of 5 chapters: Data capture and Data Entry, Operating Systems, Software Types & Programming Languages, Machine Logic and Main Storage.

Each chapter is subdivided into smaller ones, which are followed by comprehension questions in order to check understanding of the section material. For language material comprehension control check exercises were used by the authors: comprehension, translational, word-formation, sentence building exercises and retelling.

The tasks are easy to use in class or for self-study.

The authors & publishers would like to thank the teachers of English Language & Intercultural Communication of Saratov State University that took an active participation in preparation of this book & reviewer who commented on the material.

## DATA CAPTURE AND DATA ENTRY

These days  the majority of computer end-users input data  to  the computer via keyboards on PCs, workstations or terminals. However, for many medium and large scale commercial and industrial applications involving large volumes of data the use of keyboards is not practical or economical. Instead, specialist methods, devices and media are used.

The selection of the best method of data entry is often the biggest single problem faced by those designing commercial or industrial computer systems, because of the high costs involved and numerous practical considerations.

The best methods of data entry may still not give satisfactory facilities if the necessary controls over their use are not in place.

Problems of data entry

The data to be processed by the computer must be presented in a machine-sensible form (i.e., the language of the particular input device). Therein lies the basic problem since much data originates in a form that is far from machine sensible. Thus a painful error-prone process of transcription must be undergone before the data is suitable for input to the computer.

The process of data collection involves getting the original data to the "processing centre", transcribing it, sometimes converting it from one medium to another, and finally getting it into the computer. This process involves a great many people, machines and much expense.

A number of advances have been made in recent years towards automating the data collection process so as to bypass or reduce the problems.

Data can originate in many forms, but the computer can only accept it in a machine-sensible form. The process involved in getting the data from its point of origin to the computer in a form suitable for processing is called Data Collection. Before dealing with the individual stages of data collection it should be noted that data collection starts at the source of the raw data and ends when valid data is within the computer in a form ready for processing.

Many of the problems of data entry can be avoided if the data can be obtained in a computer-sensible form at the point of origin. This is known as data capture. The capture of data does not necessarily mean its immediate input to the computer. The captured data may be stored in some intermediate form for later entry into the main computer in the required form. If data is input directly into the computer at its point of

origin the data entry is said to be on-line. If, in addition, the method of direct input is a terminal or workstation the method of input is known as Direct Data Entry (DDE). The term Data Entry used in the title usually means not only the process of physical input by a device but also any methods directly associated with the input.

Stages in data collection

The process of data collection may involve any number of the following stages according to the methods used.

• Data creation, e. g, on clerically prepared source documents.

• Transmission of data.

• Data preparation, i.e., transcription and verification.

• Possible conversion from one medium (e. g, diskette) to another (e. g, magnetic tape cartridge).

• Input of data to the computer for validation.

• Sorting.

• Control — all stages must be controlled.

Not all data will go through every stage and the sequence could vary in some applications. Even today, a high proportion of input data starts life in the form of a manually scribed or typewritten document and has to go through all the stages. However, efforts have been made to reduce the number of stages. Progress has been made in preparing the source document itself in a machine-sensible form so that it may be used as input to the computer without the need for transcription. In practice, the method and medium adopted will depend on factors such as cost, type of application, etc.

Stages explained

a. Data creation. There are two basic alternatives, the first of which is the traditional method:

i. Source documents. A great deal of data still originates in the form of clerically prepared source documents.

ii. Data capture. Data is produced in a machine-sensible form at source and is read directly by a suitable device, e. g, a bar code reader.

b. Data transmission. This can mean different things according to the method and medium of data collection adopted.

i. If the computer is located at a central point, the documents will be physically "transmitted".

ii. It is also possible for data to be transmitted by means of telephone lines to the central computer.

c      Data preparation. This is the term given to the transcription of data from the source document to a machine-sensible medium. There are two parts; the original transcription itself and the verification process that follows.

Note. Data Capture eliminates the need for transcription.

d.      Media conversion. Very often data is prepared in a particular medium and converted to another medium for faster input to the computer, e.g., data might be prepared on diskette, or captured onto cassette, and then converted to magnetic tape for input. The conversion will be done on a computer that is separate from the one for which the data is intended.

e.      Input. The data, now in magnetic form, is subjected to validity checks by computer program before being used for processing.

f.Sorting. This stage is required to re-arrange the data into the sequence required for processing. (This is a practical necessity for efficient processing of sequentially organised data in many commercial and financial applications.)

g.      Control. This is not a stage as such, because control is applied throughout the whole collection.

Data-collection media and methods in outline

The alternatives are as follows.

a.      On-line transmission of data from source, e.g., Direct Data Entry (DDE).

b. Source document keyed directly into diskette (key-to-diskette) from some documents.

c The source document itself prepared in machine-sensible form using Character Recognition techniques.

d.      Data Capture Devices.

e.      Portable encoding devices.

f. Source data captured from "Tags", Plastic Badges or strips.

g.      Creation of data for input as a by-product of another operation.

*On-line systems*

The ultimate in data collection is to have the computer linked directly to the source of the data. If this is not feasible then the next best thing is to "capture" the data as near as possible to its source and feed it to the computer with little delay. Such methods may involve the use of data transmission equipment if the point of origination is remote from the

computer. The computer is linked to the terminal point (the source of data or nearby) by a telecommunication line and data is transmitted over the line to the computer system.

Data enters the terminal either by keying in via a keyboard or by a device such as one that can directly read source documents.

Such systems can involve large capital outlay on the necessary equipment, which is usually justified in terms of speed of access to the computer's data and quicker feed back of information.

If Direct Data Entry (DDE) is used then data is keyed in to a VDU or workstation by the operator. It is checked, displayed on the screen and then entered for processing. A DDE station can also be used to interrogate the computer's stored data so that data can be checked for validity at the time of input.

Some specialised on-line systems are used by the general public. Probably the most common examples are the "hole in the wall" machines, more correctly called on-line cash service tills, (or Automated Teller Machines ATMs), which are provided by many banks and building societies for cash dispensing and other services outside normal office hours.

*Key-to-diskette*

The name "key-to-diskette" comes from the days when these were special purpose systems. Today the term may be used for PC applications where data is entered onto magnetic media (typically diskettes) as an alternative to on-line systems.

*Character recognition*

The methods described so far have been concerned with turning data into a machine-sensible form as a prerequisite to input. By using Optical Character Recognition (OCR) and Magnetic Ink Character Recognition (MICR) techniques, the source documents *themselves* are prepared in a machine-sensible form and thus *eliminate* the transcription stage. Notice, however, that such characters can *also* be recognised by the human eye.

*Data capture devices* These devices are mostly special-purpose devices intended for use in particular applications. 1) Direct Input Devices.

a.    Special sensing devices may be able to detect events as they happen and pass the appropriate data directly to the computer.

b.    Voice data entry (VDE) devices. Data can be spoken into these devices. Currently they are limited to a few applications in which a small vocabulary is involved.

2) Data loggers/recorders. These devices record and store data at source. The data is input to the computer at some later stage.

A basic device, using only a keyboard for data entry, and able to transmit data, is effectively a portable terminal.

Popular attachments to both portable and static devices are the light-pen and magnetic-pen. These attachments resemble pens at the end of a length of electrical flex. More bulky hand held alternatives are sometimes called "wands". They can read specially coded data in the form of either optical marks/characters, or magnetic codes, which have previously been recorded on strips of suitable material. A common version is the bar-code reader.

*Tags*

The use of tags as a data collection technique is usually associated with clothing

retailing applications, although they are also used to some extent in other applications.

a.The original tags were miniature punched cards. Today most tags in use have magnetic strips on them instead of holes.

b. Using a special code, data such as price of garment, type and size, and branch/department are recorded on the tag by a machine. Certain of the data is also *printed* on the tag.

c Tags are affixed to the garment before sale and are *removed* at the point of sale. A t the end of the day's trading each store will send its tags (representing the day's sales) in a container to the data processing centre. Alternatively, the tags may be processed at the point of sale.

d.At the centre the tags are *converted* to more conventional diskette or magnetic tape for input to the computer system. e.Note that data is "captured" at the source (point of sale) in a machine-sensible format thus needs no transcription and can be processed straight away by the machine

*Bar-coded and magnetic strips*

Data can be recorded on small strips, which are read optically or magnetically. Optical reading is done by using printed "bar codes", i.e., alternating lines and spaces that represent data in binary. Magnetic reading depends on a strip of magnetic tape on which data has been encoded. The data are read by a light-pen, magnetic-pen or wand which is passed over the strip. Portable devices are available that also include a keyboard. An example of their use is in stock recording; the light pen is used to read the stock code from a strip attached to the shelf, and the

quantity is keyed manually. The data are recorded on a magnetic tape cassette. This technique is also used at check out points in supermarkets. Goods have strips attached and stock code and price are read by the light pen. The data thus collected are used to prepare a receipt automatically, and are also recorded for stock control purposes.

**Comprehension check**

**Answer the questions**

1. What is the basic problem of data entry? How can this problem be avoided?
2. What are the basic alternatives in data creation?
3. What stage is applied throughout the whole process of data collection?
4. Which data-collection methods give an opportunity to create documents in a machine-sensible form?

**Find the English equivalents of the following terms:**

Обрабатывать данные, побочный продукт, вводить данные, затраты (издержки), сбор данных, регистратор данных, исходный документ (первоисточник информации), преобразовывать), необработанные данные, магнитная полоса, устройство считывания штрихового кода, достоверные данные, передача данных, подвергаться, реальный (выполнимый), запрашивать данные, необходимое условие, приставка, набирать на клавиатуре, подверженный ошибкам.

**Fill in the table. Use 10 words in the sentences of your own. Use the dictionary to help you when necessary.**

| Noun | Verb | Adjective |
|---|---|---|
| | to transcribe | |
| verification | | |
| selection | | |
| | | valid |
| | to justify | |
| attachment | | |
| | | considerate |
| interrogation | | |
| | to resemble | |

**Use the following collocations in the sentences of your own.**

Terminal: dial-up terminal, data-entry terminal, idle terminal, inquiry terminal, master terminal, job-oriented terminal, network terminal, orphan terminal, smart terminal.

Version: backup version, down-sized version, incorrect version, released version, marketed version, demo version.

Attachment: file attachments.

**Translate the sentences into English, using the vocabulary of the** passage:

1. Новые методики помогут тебе обойти эту проблему.

2. Как только ты получишь какую-нибудь информацию о его новой работе, позвони мне.

3. Мы подверглись тщательному обыску, прежде чем нам позволили встретиться с ним.

4. *Я* не думаю, что использование этого прибора оправдано в данной ситуации.

5. Эти насекомые походили на пауков, но были намного больше по размеру.

6. Это пятно очень трудно удалить.

7. Они часто встречались и очень много разговаривали.

8. Она сказала, что мнение общественности необходимо учитывать.

9. Она очень много в него вложила, но не получила никакой отдачи. Ю.Мне сказали, что он производит отбор персонала на этом предприятии.

**Write a summary of the passage using the following vocabulary**

- Machine-sensible
- To originate
- To undergo
- To automate
- To vary
- To go through stages
- To make efforts
- To be linked directly to...
- By means of
- Version.

# OPERATING SYSTEMS

Introduction

Why operating systems were originally developed

Over many years the increased processing speed of the processor and its massive problem-solving capability brought about the need for more sophisticated modes of *operating* computers.. What is required is a "super controller" to ensure that the vast facilities are used to optimum advantage. It became clear that the job could only be undertaken by some form of *internally* stored program. This became known as an operating system.

Definition

From what has just been stated, an operating system may be seen as a suite of programs that has taken over many of the functions once performed by human operators. The role of the operating system is therefore one of resource management. The primary resources it manages are:

a. processors,

b. storage,

c  I/O devices,

d. data,

e.  programs.

An operating system is a suite of programs that takes over the operation of the computer to the extent of being able to allow a number of programs to be run on the computer without human intervention by an operator.

Basic organization

It can be seen from what has been said so far that the operating system controls the way software uses hardware. This control ensures that the computer not only operates in the way intended by the user but does so in a systematic, reliable and efficient manner.

1. The Operating System's place in the organisation of the overall system. Part of the operating system remains in main storage permanently during the running of the computer. This part is called the Kernel (or Supervisor or Executive) and as the name suggests is the "controlling" part of the operating system. It controls the running of all other programs. The remainder of the operating-system programs are stored on a direct-access storage device (preferably) from which any particular one will be "called" into main storage by the kernel when required.

On very small computers the operating system may take the form of a rudimentary control program sometimes called a monitor, an unfortunate name because of potential confusion with a VDU monitor.

On many very small microcomputers the kernel is stored permanently in ROM and starts execution the moment the computer is turned on.

On most other modern computers the kernel is not in main storage when the machine is switched on. The system must be "booted up". This sometimes involves pressing special "boot buttons", keys or switches, which cause the hardware to load the kernel into main storage from a predetermined position on a disk.

The functions of an operating system

The functions may include:

a. The scheduling and loading of programs, or subprograms, in order to provide a continuous sequence of processing or to provide appropriate responses to events.

b. Control over hardware resources, e.g., control over the selection and operation of devices used for input, output or storage.

c. Protecting hardware, software and data from improper use.

d. Calling into main storage programs and subroutines as and when required.

e. Passing of control from one job (program) to another under a system of priority when more than one application program occupies main storage.

f. Provision of error correction routines.

g. Furnishing a complete record of all that happens during processing (usually in the form of a printed log).

h. Communication with the computer operator usually by means of the console typewriter. The operator communicates with the operating system by means of a JCL (Job Control Language), which is translated and executed by an interactive command interpreter.

Methods of operation and modes of access

Operating systems are often described in terms of the methods by which they *operate the system* and the modes of system access they provide. The common alternatives are introduced here.

a. Multiprocessing. This is the name for the situation that occurs if two or more processors are present in a computer system and are sharing some or all of the same memory. In such cases two programs may be processed at the same instant. In the remainder of this discussion assume just one processor is used unless otherwise stated.

b. Multiprogramming. This occurs when more than one program in main storage is being processed apparently at the same time. This is accomplished by the programs taking turns at short bursts of processing time.

c Batch processing. The job (program + data) is not processed until fully input. The jobs are entered and stored on a disk in a batch queue and then run one or more at a time under the control of the operating system. A job may wait in a batch queue for minutes or hours depending on the work load. No amendments are possible during processing.

d. Remote job entry refers to batch processing where jobs are entered at a terminal remote from the computer and transmitted in to the computer.

e. Interactive computing. This occurs if the computer and terminal user can communicate with each other.

f. Conversational mode. This is interactive computer operation where the response to the user's message is immediate.

g. Multi-access. This occurs if the computer allows interactive facilities t more than one user at a time.

h. Time-sharing. Processor time is divided into small units called time slices and shared in turn between users to provide multi-access.

The following facilities are usually made available by a time-sharing operating system.

1. The programmer is informed of errors in his source program or execution of his program immediately.

2. The programmer is able to make corrections to his source program on-line, i.e., editing is possible.

3. Debugging (i.e., removing the faults (bugs) from a program). This is done by the programmer being able to have variable values printed at various points in the execution and by amending values at the same time.

i. Real-time system. A real-time system is a computer system that is capable of processing data so quickly that the results are available to influence the activity currently taking place. There is often a need for multiprocessing and a front-end processor in these systems.

Choice of operating system

Choice of operating system. The applications for which a computer is needed largely determine the choice of hardware and accompanying software. The operating-system supplier will need to consider these factors:

a. The hardware provision and basic design of the computer.

b. The applications intended for the computer.

c The method of communication with the computer, eg, many or few peripherals,

d. The method of operating the computer.

Types of operating system

*Single-program systems.* The majority of small microcomputer-based systems have operating systems, which normally only allow one user program to be in main storage and processed at a time, i.e., there is no multiprogramming of user programs. *Simple batch systems.* These are systems that provide multiprogramming of batch programs but have few facilities for interaction or multi-access. Many commercial computer systems in use during the 1960s and early 1970s were of this type.

*Multi-access and Time-sharing systems.* The majority of operating systems fall into this category, but there is a wide range of complexity in such systems.

On mainframes, operating systems are normally specific to the particular machine and manufacturer.

*Real-time systems.* The operating system has to cater for the type of real-time system being used. The three types are given here in order of increasingly fast response time.

a. A more complex multi-access time-sharing system where each user has a largely independent choice of system facilities, e.g., each using a different language.

b. Commercial real-time systems in which there is essentially one job, such as handling booking, and the multi-access user has a clerical rather than programming function. These systems often make use of extensive data bases.

c Process control systems, e.g., a system to control the operation of a chemical-factory plant. Response to changes must be as fast as possible and reliability is essential. At the other extreme are the embedded Real-time control systems used in microprocessor-based instruments and monitoring devices, e.g., the control system in a service station petrol pump.

Operating system languages

The three types of language to be considered here are:

a. Command languages.

b. Job-control languages.

c  Languages in which to write operating-system software.

Command language. A programming language used for communications with the operating system is called a command language. Some command languages provide no more than a set of single-line commands, which may be keyed in by a user, e.g., individual commands to print, compile, load, execute, copy or delete a named file. Other command languages also provide control structures so that commands may be selected or repeated according to specified conditions. An example of the latter is the shell language on the UNIX operating system. Job Control Language (JCL). This is a special command language that is used to identify jobs and state their requirements to the operating system. Languages in which to write operating system software vary considerably. It is common to find some parts of the operating system's software written in assembly language because of the machine-orientated nature of the software or for reasons of efficiency. However, in order to aid operating-system portability and improve system programmer productivity a number of higher-level languages are used. These are often variations on high-level languages or are special-purpose languages.

Processes

A concept fundamental to the understanding of how an operating system works is that of a process.

A process may be thought of as the *execution of a program,*. The process may be in a state where it is waiting for the processor to continue execution or it may be in the running state. If running is suspended it will be necessary to save details of the point reached in the program and the state of variables, registers, etc. For re-entrant programs the data associated with each process must be separately managed, just as it would be necessary for each reader to keep track of the place reached if a book being read was shared.

Multiprogramming in batch

The operating system will try to optimise available to it using the following methods. *Job priority.* It is also advantageous to assign a priority to the job before it is submitted, for example, by a command switch. The two factors influencing this are:

a. The relative proportion of I/O operation and processing.

1. A job requiring a large amount of I/O operation will be given high priority because such a program will allow other programs more opportunity to use processor time.

2. A job requiring a large amount of processor time would be given low priority because if given high priority it would prevent *I/O* programs from obtaining the small amount of processor time they need.

b. A secondary factor is the urgency with which the program must respond to peripherals, since data can be lost if some devices such as magnetic tape units are not serviced in time.

Jobs requiring a large amount of peripheral time are called peripheral bound or I/O bound. Jobs requiring a large amount of processing time and little I/O are called processor bound, or CPU bound.

*Scheduling.* There are two types of scheduling in a batch-operating system.

a. Job schedule. This system software routine works out a "priority index" for each job based on such factors as:

1. Processor time required by the job.

2. I/O time required by the job.

Resources available on-line, e.g., free memory and free I/O devices.

3. The deadline by which the job must be completed.

b. Processor schedule. This system software routine is entered frequently to decide "WHAT TO DO NEXT". It works out the hardware resources currently demanded by each job and checks what resources are available. It then gives processor time to the most appropriate job according to a scheduling strategy such as: "Process the highest priority job that can use available resources". The job given processor time will run until 1. It has ended, or 2. an interrupt occurs, e.g., I/O request, operator break-in or run-time error, or 3. an interrupt is requested by the job, e.g., for output.

*Passing control from one job to another under a scheduling scheme.* This transfer of control is handled by the kernel as follows: a. The cause of an interrupt may be:

1. External, e.g., an input device requesting service, or

2. internal, e.g., a programming error that causes an exit from the compiler.

In either case control is transferred to the *kernel.*

b. In the case of an external interrupt the kernel might transfer control to a routine to handle data input, taking it from a processor-bound program.

c    In the case of an internal interrupt the kernel might terminate the job and pass control to the most suitable program under the given scheduling scheme.

The general work of operating systems

It is helpful at this stage to look through the general work common to most operating systems. The main headings are:-

a. I/O control

b. Interrupt                    handling

c   Error correction

d. Device allocation

e. File security and management

f.  Log of events and operator communication

g. Loading programs and subroutines.  Storage management

  I/O control. This will consist of a set of programs for controlling the input from and output to the various devices. These programs form part of the kernel and are entered via the kernel when an interrupt occurs, e.g., one such program might set up and read blocks from a tape.

Interrupt handling. This is one function of the kernel. When an interrupt occurs, control is passed to the kernel, which determines the cause of the interrupt and transfers control to the most appropriate program.

The operating system should be able to handle errors that occur as part of the normal operating activity. For example, if data is lost during transfer because of late service of an interrupt, a routine should be available to recover the data by setting up the read process again.

Allocation of devices. Several jobs may request the same output media. If more than one output device is available for that medium, then some programs will be allocated one device and other programs another device. If subsequently the queue for one device is longer than for another then reallocations are made. To do this track must be kept of the allocations made. File security and management. This involves these main activities:

a. Preventing illegal access or amendments to user or system files.

b. "Dumping" the contents of files from time to time in case of future loss, e.g., disk files may be transferred to tape and then stored off-line.

c   Keeping track of file descriptions and locations.

  Furnishing a log of events.

a. Most of the information will be presented to the console and provided by a program, which keeps track of job execution and hardware use.

b. Information may also be furnished as part of the user's job output either as a result of errors or because of job-control requests. Further contributions to the console log come from communication with the operator.

Communication with the operator. The operator has the responsibility of dealing with errors missed by the operating system, and in the event of operating-system failure the operator must re-establish it.

Loading programs and subroutines. Main storage is not usually able to hold all the software that is being used so transfers between main storage and direct-access storage are frequently being made.

Storage management

The operating system may organise main storage into blocks of convenient size called partitions. It can then allocate storage space by partitions rather than by piecemeal allocation by individual locations.

a. In order to place jobs in whatever partition of main storage is available the programs must be relocatable. Their absolute (hardware) address is set by the kernel when it adds the base address of the partition to the relative address of each instruction.

b. The kernel may be able to change continually the allocation of programs by repeated re-applications of the process described in (a). This is called dynamic allocation of main storage.

*Tasking, Multi-tasking and Segmenting.*

a. Today, the term multi-tasking is most often taken to mean the multiprogramming of *user programs.* On a single-user computer *without multi-tasking,* such as MSDOS, the multiprogramming is primitive in that there is typically a very limited number of programs running together. These are: the operating system, *one* interactive user program and *possibly* a background (i.e., non-interactive) program running a printer queue. In a multi-tasking system there can be a number of interactive user programs being multi-programmed together with the operating system and probably some background tasks too.

b. True multi-tasking is pre-emptive which means that if a process is interrupted it will not necessarily be resumed immediately after the interrupt has been serviced because other processes may have a higher priority. Modern general purpose minicomputers and mainframes have pre-emptive multi-tasking operating systems. In the case of non pre-emptive multi-tasking a single user process, if interrupted, will normally be resumed once the interrupt has been serviced and may therefore gain an unfair share of the processors time. It has the general appearance of

multi-tasking to the user but provides a less smoothly balance performance between tasks. Non pre-emptive multi-tasking is sometimes called pseudo multitasking.

c On some minicomputers and mainframes there are more sophisticated forms of multi-tasking. In such cases a job may be broken down into stages called tasks. It may be possible to treat these tasks as separate programs to the extent that they may be multi-programmed with each other. This is called tasking.

d. A job may be divided up into convenient storage units called segments. (A segment may well correspond to a task.) These segments may be organised so that only those segments currently requiring processing are held in main storage. The rest are held on direct-access storage. This is known as segmenting. As each new segment is brought into main store it is said to overlay the segment it replaces. When segments of different programs are multi-programmed the method is called interleaving.

Virtual Storage. A combination of segmenting and dynamic allocation of main storage can make it possible for the programmer to write a program (which can be broken into segments) with little regard to the main storage available. The segments of the program will be brought into main storage from special files on direct-access storage and dynamically allocated a block of main storage (a partition). Such a storage management arrangement is known AS VIRTUAL storage since main storage can be regarded as larger than its real size. The special files on direct access storage used to hold the programs, or segments of programs, not currently in physical memory are called page files or swap files. When a program, or program segment, is transferred between main storage and such a file it is said to be "paged" or "swapped" in or out of memory.

**Comprehension check.**

**Answer the questions**

1. Why were operating system originally developed?
2. What is the definition of an operating system?
3. What are the primary resources an operating system can manage?
4. What are the functions of an operating system?
5. What are the types of operating systems?
6. What are the languages used to write an operating system?
7. What is a virtual storage?
8. In what way is a real-time operating system different from other ones?

9.What are the advantages of time-sharing operating systems?

10.What do shareable resources include?

**Translate these terms from English into Russian:**

- Storage
- I/O device
- Multiprogramming
- Multiprocessing
- Batch processing
- Real-time system
- Command language
- Job priority
- Scheduling
- Interrupt
- Log of events

**Translate the sentences into English using the vocabulary of the passage**

1.Исходя из того, что было сказано, операционную систему можно рассматривать как набор программ, которые приняли на себя многие функции, ранее выполняемые операторами.

2.Язык программирования, используемый для связи с операционной системой, называется командным языком.

3.Если во время выполнения процесса возникает ошибка, и она определяется операционной системой, процесс будет приостановлен.

4.Когда происходит прерывание, контроль переходит к ядру, которое определяет причину прерывания и передает управление наиболее подходящей программе.

5.Операционная система должна уметь справляться с ошибками, являющиеся результатом обычной работы операционной системы.

6.Операционная система способна организовывать память в блоки удобного размера, называемые ячейками.

7.Ядро может постоянно изменять расположение программ путем повторного применения процесса, указанного ранее.

8.Когда сегменты различных программ подвергнуты мультипрограммированию, такой метод называют чередованием адресов памяти.

9.Специальные файлы в памяти прямого доступа, используемые для того, чтобы держать программы или сегменты программ, но не находящиеся в настоящее время в физической памяти, называются

файлами регистрации или файлами подкачки.

10.Общие ресурсы включают процессоры, файлы, доступные только для чтения, и области памяти, защищенные от изменений.

**Use the following collocations in the sentences of your own (make up 15-20 sentences).**

<u>Batch</u>: card batch, job batch, out-of-balance batch, remote batch.

<u>Factor</u>: activity factor, certainty factor, coding load factor, complexity factor, feedback factor, fault factor, human factor, work load factor.

<u>Strategy</u>: fault-tolerance strategy, first-look strategy, interconnection strategy, layout strategy, safe strategy, sequential strategy.

<u>Block</u>: control block, dead block, decision block, file access block, fixed block, labeled block, message block, memory block, page control block, queue control block, spare block, storage lock, block of decomposition.

<u>Console</u>: alternate console, auxiliary console, desk control console, local console, message display console, primary console, recovery console, remote console, secondary console, supervisory console, test console, user console, visual operator console.

<u>Segment</u>: common segment, database segment, delimiter segment, exclusive segments, header segment, inclusive segments, masked segment, paired segment, root segment.

**Write a summary of the passage, using the following vocabulary:**

- Operating system
- Set-up time
- Manual intervention
- Suite
- Storage
- I/O device
- Data
- Software
- Hardware
- Kernel
- Multiprocessing
- Multiprogramming
- Batch
- Real-time

# SOFTWARE TYPES AND PROGRAMMING LANGUAGES

Types of software

The two basic types of software are:

a. Systems software

b. Applications software.

Systems Software. The user of a computer has at his or her disposal a large amount of software provided by the manufacturer. Much of this software will be programs that contribute to the control and performance of the computer system. Such programs are given the collective name systems software. Any one of these programs is a systems program.

Applications software. Applications programs may be provided by the computer manufacturer or supplier, but in many cases the users produce their own applications programs, called user programs. Most applications programs can only work if used in conjunction with the appropriate systems programs.

Further subdivision. A more detailed subdivision is as follows:

a. Operating systems and control programs

b. Translators

c   Utilities and service programs

d. Data Base Management Systems (DBMS)

e. User applications programs

f.  Applications packages

Operating systems and control programs

Computers are required to give efficient and reliable service without the need for continual intervention by the user. This requirement suggests that computers should monitor and control their own operation where possible. Control programs and operating systems are the means by which this is achieved. Such programs control the way in which hardware is used.

Under many circumstances at least two programs are within main storage. One may be an applications program, the other will be a control program, which monitors, aids and controls the applications program.

On small computer systems, such as microcomputers, the control program may also accept commands typed in by the user.   Such a control program is often called a monitor.

On larger computer systems it is common for only part of the monitor to remain in main storage. Other parts of the monitor are brought into

memory when required, i.e., there are "resident" and "transient" parts to the monitor.

The resident part of the monitor on large systems is often called the Kernel, Executive or Supervisor program.

The kernel is one of a suite of control programs that are able to allow a number of applications programs to run concurrently and in sequence without the intervention of the user. Such a suite of control programs is called an operating system, but a more precise definition is now given.

An operating system is a suite of programs that takes over the operation of the computer to the extent of being able to allow a number of programs to be run on the computer without human intervention by an operator.

Translators

The earliest computer programs were written in the actual language of the computer. Nowadays, however, programmers write their programs in programming languages which are relatively easy to learn. Each program is translated into the language of the machine before being used for operational purposes. This translation is done by computer programs called translators.

Utilities and service programs

Utilities, also called service programs, are systems programs that provide a useful service to the user of the computer by providing facilities for performing common tasks of a routine nature. Common types of utility programs are:

a. Sort

b. Editors

c File copying c Dump

e. File maintenance

f. Tracing and debugging.

Sort. This is a program designed to arrange records into a predetermined sequence. A good example of the requirement for this service program is the need for sorting transaction files into the sequence of the master file before carrying out updating. Sorting is done by reference to a record key.

Editors are used at a terminal and provide facilities for the creation or amendment of programs. The editing may be done by the use of a series of commands or special edit keys on the keyboard. If, for example, a source program needs correction because it has failed to compile properly, an editor may be used to make the necessary changes. File

copying This is a program that simply copies a file on the same medium to make a duplicate or from one medium to another, e.g., from one disk to another (also called media conversion).

Dump. The term "Dump" means "copy the contents of main storage onto an output device". This program is useful when an error occurs during the running of application programs. The printed "picture" of main storage will contain information helpful to the programmer when trying to locate the error. It is also used in conjunction with a CHECKPOINT/RESTART program. This program stops at various intervals during the running of an applications program and dumps the contents of main storage. If required, the checkpoint/restart program can get the applications program back to the last checkpoint and restart it with the conditions exactly as they were at the time.

File maintenance. A program designed to carry out the process of insertion/deletion of records in any files. It can also make amendments to the standing data contained in records. File maintenance may also include such tasks as the reorganisation of index sequential files on disk.

Tracing and debugging. Used in conjunction with the debugging and testing of application programs on the computer. Tracing involves dumping details of internal storage (e.g. the value of a variable) after obeying specified instructions so that the cycle of operations can be traced and errors located. Debugging is the term given to the process of locating and eliminating errors from a program.

<u>User applications programs</u>

User applications programs are programs written by the user in order to perform specific jobs for the user. Such programs are written in a variety of programming languages according to circumstances, but all should be written in a systematic way.

For many applications it is necessary to produce sets of programs that are used in conjunction with one another and that may also be used in conjunction with service programs such as sort utilities.

For some applications it may be necessary to compare available software or hardware before writing the applications software. For example, the user may wish to compare the performance of two different COBOL compilers. Such a comparison may be made using a benchmark.

Benchmarks. A benchmark is a standard program, or assembly of programs used to evaluate hardware or software. For example, a benchmark might be used to:

a. Compare the performance of two or more different computers on identical tasks.

b. Ascertain if the configuration provided by a manufacturer performs according to his claims.

c  Assess the comparative performance of two operating systems. <u>Types of programming languages</u>

There are three types of programming language:

a. Machine language itself

b. Low-level                                                                                  languages

c  High-level languages

<u>The development of high-level languages</u>

The development of high-level languages was intended to overcome the main limitations of low-level languages, which are:

a.  Program writing is a relatively time-consuming business for the programmer because the assembly process produces machine instructions on a ONE-for-ONE basis.

b. Low-level languages are machine orientated, each conforming to the instruction set of the machine on which they are used and therefore restricted to use on that machine.

High-level languages are intended to be machine independent and are problem-orientated languages (POLs), i.e., they reflect the type of problem solved rather than the features of the machine. Source programs are written in statements akin to English, a great advance over mnemonics.

The first high-level language to be released was FORTRAN (FORmula TRANslation) in 1957. Over the next few years many other high-level languages appeared and established themselves. In fact, new high-level languages have been coming into existence ever since. Most, but not all, have disappeared into obscurity.

The machine independence of high-level languages means that in principle it should be possible to make the same high-level language run on different machines and then write programs in that language that are portable. Such portability has a number of advantages.

a. A user of a program can change to a newer or bigger computer without the need to rewrite programs.

b.   Users of different computers may be able to share or exchange programs and thereby reduce costs.

c An organisation producing software for sale can sell the same program to users of different computers without the need to rewrite the program for each type of computer. Again, this reduces costs.

In practice, there are a number of obstacles to such portability. Perhaps the most serious obstacles are those resulting from variations that occur between different versions of each language. Each different *dialect* is usually the deliberate creation of **a** computer manufacturer, who will try to attract new customers with a "better" product and who will then keep existing customers because their programs will no longer be portable to the computers of other manufacturers.

**Comprehension check Answer the questions**

1. What is the difference between the two basic types of software?
2. What is an operating system?
3. What are translators used for?
4. Characterize common types of utility programs
5. What are the types of programming languages? What necessitated the development of high-level languages?

**Find the English equivalents of the following terms:**

Язык программирования, точное определение, выполнить программу, подробный, поправка, правка, оперативное запоминающее устройство, вставка, портативный, производитель, поставщик, программист, внутреннее запоминающее устройство, переменная, уничтожить ошибку, средство (способ), удостовериться, следовать инструкциям, носитель.

**Fill in the table. Use 10 words in the sentences of your own. Use the dictionary to help you when necessary.**

| Noun | Verb | Adjective |
| --- | --- | --- |
| | to eliminate | |
| | | accessible |
| intervention | | |
| | to maintain | |
| | | convertible |
| | to reduce | |
| | | valuable |
| occurrence | | |
| | to release | |

**Use the following collocations in the sentences of your own (make up 15-20 sentences).**

Application: customized application, business application, direct access application, host application, multiuser application, slave application, host application, multiuser application, and slave application.

27

Sequence: data sequence, ascending sequence, decreasing sequence, illegal sequence, inverted sequence, instruction sequence, key sequence, trigger sequence.

Facilities: advanced facilities, change facilities, browse facilities, debugging facilities, fault tracing facilities, help facilities, input/output facilities, security facilities; to provide facility for.

Medium: information-carrying medium, machine-readable medium, input medium, output medium, portable data medium, recording medium, storage medium, transmission medium.

Error: accidental error, accuracy error, admissible error, common error, detectable error, external error, fatal error, initial error, misuse error, out of memory error, recurrent error, setup error; to make/commit an error, to correct/rectify an error, to compound an error.

Data: actual data, bad data, available data, chain data, compatible data, cumulative data, current data, derived data, downloaded data, encrypted data, engineering data, field data, incoming data, integrated data, lost data; to collect data, to evaluate data, to feed in data, to process data, to retrieve data, to store data.

**Translate the sentences into English, using the vocabulary of the passage**

1. Я не могу сейчас взяться за этот проект, потому что это будет отнимать много времени, а я сейчас очень занят.

2. Ни при каких обстоятельствах ты не должен помогать ему.

3. Ты недостаточно компетентен, чтобы оценить его познания.

4. В твоем распоряжении самое современное оборудование, и все равно ты не можешь решить эту проблему.

5. Доступ к этой информации ограничен.

6. Если ты будешь работать более эффективно, ты вскоре достигнешь своей цели.

7. Всегда полезно располагать достаточной информацией.

**Write a summary of the passage using the following vocabulary**

- Systems/applications software
- Operating system
- Main storage
- Intervention
- To provide facilities
- In order to
- To overcome the limitations of

- To be problem-orientated
- To be a great advance over...
- To come into existence/to disappear into obscurity.

# MACHINE LOGIC

Here we will deal with how the computer handles binary data and instructions at what may be called the digital logic level of the machine.

Data and instructions are transmitted between the various parts of the processor or between the processor and peripherals by means of pulse trains. Various tasks are performed by passing pulse trains through "electronic switches" called gates. Each gate is an electronic circuit that may have provision for receiving or sending several pulses at once. Each gate may be regarded as a "black box" that controls the flow of pulses in a particular way.

Each gate normally performs some simple function (operation), e.g., AND, OR, NOT, and for this reason gates are often called logic elements. It is the logic operations of gates which will concern us here , and we will not consider how they perform electronically, nor will we consider how they are made. The following background information should suffice.

Background information

The key points are as follows:

a. In modern computers a large number of components are made from small pieces of semi-conductor material called silicon chips. A single chip will contain several components (e.g., all the components for one gate) and because of this electronic property may be referred to as an integrated circuit (1C). 1Cmemory is also called semi-conductor memory.

b. Chips may be classified according to size as follows:

i. Small Scale Integration (SSI) — containing at most 10 gates.

ii. Medium Scale Integration (MSI) - containing between 10 and 100 gates.

iii. Large Scale Integration (LSI) - containing between 100 and 100 000 gates.

iv. Very Large Scale Integration (VLSI) - containing more than 100 000 gates.

NB. This classification is not too strictly adhered to. For example, a circuit with 12 simple gates might be classified as SSI.

c.     To gain an idea of what these sizes mean we may note that typical microprocessor-units (MPUs), and memory chips used with them, are examples of LSIs. Many of the devices such as adders, decoders,

multiplexers, etc, may be made using MSI, or from a combination of SSI components.

d. LSI design and manufacture is most economic if large numbers of chips are to be used. Therefore, special-purpose devices that could be manufactured using LSI maybe constructed from common MSI components instead if the numbers required do not make LSI manufacture economic. General-purpose LSIs may also be used in such cases. One type, the PLA (Programmed Logic Array), may be specially tailored to a particular purpose in a one-off operation broadly comparable to the programming of a PROM.

e.To function properly, all chips require an adequate power supply. The operation of used to provide pulses to the chip. The operations that a chip performs on the inputs it receives may either depend upon the level of the input signal (0 or 1) or upon transitions in level (0 to 1 or 1 to 0).

<u>Logic diagrams</u>

In logic diagrams *gates* are represented by *symbols* and *inputs* and *outputs* are represented by *arrowed lines* <u>labelled by</u> letters.

Example. In this diagram an AND gate (AND element) has two input signals, ЛT and Y, and gives one o<u>utput</u> signal corresponding to X.Y.

ЛLY ,

AND

There is no need to write AND inside this AND symbol once it has been learned.

Simplified symbols may be used in which the name of the gate is written inside a circle as shown here.

X, X.Y

The $\bigcirc$ (AND) functions of a gate corresponds to logic operations and so they may be defined by a truth table.

Example.

| X | Y | X.Y |
|---|---|-----|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

"1" means a "high" signal. "0" means a "low" signal. The AND gate will only output a "high" signal if it receives both X and *Y* "high" signals at the same time.

<u>Solving problems in logic design</u>

In designing a logic circuit we aim to reduce the number of gates used, thereby reducing cost .The method is outlined here.

Steps in the solution.

a. Start with a well-defined description of the problem, e.g., identify exact inputs and outputs.

b. Select symbols to represent the input and output variables.

c. Form a truth table in terms of these symbols referring to the problem definition throughout.

d. Form logic expressions from the truth table giving output variables in terms of input variables.

e. Try to simplify the output expressions using Karnaugh maps or algebra.

f. Produce a logic diagram for the simplified outputs. Build up the diagram using ANDs, ORs, or NOTs and work "backwards" from output to input. The latter is not essential but the method helps later on.

**Comprehension check**

**Answer the questions**

1. What are the functions of pulse trains and how are they connected with "electronic switches"?

2. Which other names of electronic switch do you know? How does it operate?

3. How can you classify silicon chips? Is it obligatory to strictly adhere to this grading?

4. Describe the logic diagram given above. What is the difference between positive and negative logics?

5. What steps can you use to reduce cost in logic design?

**Find the English equivalents of the following terms:**

обрабатывать двоичные данные; передавать инструкции; внешние устройства; пропускать серию импульсов; логический элемент \ вентиль; электронная схема; полупроводниковый материал; кремниевый чип; интегральная схема; строго придерживаться; сумматор; устройство специального назначения; спроектированный; источник питания; отмечать \ обозначать; упрощенный символ; соответствовать; истинностная таблица; логический элемент с выходным напряжением высокого уровня; разработка алгоритмов; сокращать затраты.

**Translate the sentences into English, using the vocabulary of the passage:**

1.Думаю, для постройки дома этого количества кирпича будет

32

достаточно.

2.В каждом радиоприемнике есть встроенные (in-built) средства, которые получают и отсылают сигналы.

3.Телевизор работал нормально до тех пор, пока папа не решил его "починить"

4.К нему почему-то относились как к хакеру, хотя он таковым не являлся.

5.Самым ценным качеством компьютера я считаю его многофункциональность.

6.В соответствии с последними подсчетами моя библиотека вмещает около 400 книг.

7.Чтобы понять его мысль, мне понадобилось какое-то время.

8.Для написания программ требуется соответствующая подготовка.

9.Изменение уровня сигнала на схеме было представлено пунктирной линией(dotted line).

10.Всю жизнь он стремился к лучшему и только под конец двинулся в обратном направлении.

**Write a summary of the passage using the following vocabulary:**

• To transmit
• Pulse trains
• Gate
• To perform function
• Semi-conductor material
• To be constructed from
• Power supply
• Input and output signals
• To depend upon
• To be defined by
• Variables
• In terms of
• To identify
• Logic expression

# MAIN STORAGE

Introduction

Main storage is an important element of a computer because it holds data and program instructions for use by the processor. The general features of main storage are described here. Another very appropriate name for main storage is Main Memory, because, like the human memory, it is able to retain or store information that may later be recalled or accessed. *Features of Main Storage*

a.  Its operation is wholly electronic, and consequently very fast and reliable. In the most modern computers the electronic memory circuits are also highly miniaturised.

b.  Data is almost instantly accessible from main memory because of its electronic operation and close proximity to the processor. For example, an item of **data** maybe retrieved from main storage in much less than one-millionth of one second. For this reason main storage has yet another name, immediate access storage.

c Data *must* be transferred to main storage before it can be processed by the processor. High access speeds then contribute to fast processing.

*Uses of Main Storage.*

Ideally, main storage would be used to store all data requiring processing, in order to achieve maximum processing speeds. However, there are practical limitations on the size of main storage that processors can easily be made to operate with. Main storage is also relatively expensive, although its price has come down in recent years as the manufacturing methods have improved. So, the practical solution is to limit the size of main storage and supplement it with less expensive backing storage. Thus, limited capacity, coupled with the need to hold data in main storage for processing, results in main storage being *used as* a short-term memory.

It stores
1. Instructions awaiting to be obeyed,
2. Instructions currently being obeyed,
3. Data awaiting processing,
4. Data currently being processed,
5. Data awaiting output.

An area of main storage being used for data currently being processed is called a working storage area and an area being used for data awaiting processing or output is called a buffer area.

The operational features of main storage

To understand the principles of how data is placed into main storage we will imagine it to be arranged like a set of boxes, and deal with the physical details of main storage later. The boxes are numbered from zero upwards so that each box can be identified and located. What is usually called a "location in main storage" corresponds to one of our boxes, and the "location address" corresponds to the number of the box. Once data is stored in a location in main storage it remains there until it is replaced by other data. Data placed into the same location will destroy what was there previously (rather like the latest recording made on a tape recorder destroying the previous one). Accessing and fetching data from main storage is really a copying action which does not result in the data being deleted from main storage (just as playing a tape recorder does not erase the tape).

*Random Access.* It is possible to fetch data from the locations in main storage in any order, and the time taken does not depend on the position of the locations. We summarise this by saying we have random access to data in main storage. This is also known as direct access.

*Volatility.* Depending on its physical characteristics, main storage may be either volatile, in which case data is lost once the power is turned off, or non-volatile, where no such data loss occurs.

*Details of a single location.* Each location in main storage consists of a set of tiny devices each of which can be in one of two states at any one time (e.g., On or Off). The two states of each device are used to represent binary 0 (Off) and 1 (On), and hence the complete set of two-state devices in each location provides a method of coding data, rather like that used with input media such as magnetic tape. We may thus talk of a location in main storage as being so many bits long. For a given computer the number of bits in each location will be the same fixed number for all locations, e.g., *each* location could contain 16 bits, and each location would be addressable.

Bytes and words

Each location in main storage is usually said to hold a unit of data called a word. Words can often be subdivided into smaller units called bytes. The real situation is much more complicated than this because of the many different approaches    to computer design. The next few paragraphs discuss common design
approaches.

*Fixed Word-length Computers (Word Machines).* In a fixed length computer one word = one location in main storage, i.e., data is

transferred into main storage one location at a time. The number of bits in each location (i.e. in each word) is known as the word length and depends on the make and model of computer. Typical word lengths are, 8, 16, 32 and 64 bits. Several characters might be placed or "packed " into one word.

Microcomputers are normally fixed word-length machines and the smaller older ones often have a word length of 8 bits. A single storage location of such a machine is large enough to hold one ASCII character, one BCD character, one EBCDIC character or two BCD numeric values. A microcomputer using this method of storage
is called an 8-bit microcomputer (see Fig. 6.2).

On larger microcomputers the most common size for a memory location is 16 bits and computers with memory locations of this size are called 16-bit computers.

Currently most PCs are 16-bit computers but there is a trend towards 32 bits. A few minicomputers are 16-bit machines too. A single storage location on a 16-bit computer will be able to hold two characters (e.g. ASCII or EBCDIC) but will normally "pack" characters two to a location for that reason (see Fig. 1). Such computers usually have facilities for accessing the right half and left half of each word separately in order to handle individual characters.

32-bit storage locations are found on some of the more modern microcomputers, on most minicomputers and on the smaller mainframe computers. Such 32-bit computers normally have a variety of ways of accessing all or part of each memory location so that, for example, packed characters may be accessed singly if required.

In the case of mainframes there is a greater variety of ways of organising storage with alternatives to the fixed word-length machines being quite common. The reason is that in general, storage space tends to be wasted by using fixed-length units of storage for variable-length data, e.g., having to use a 16-bit word to store one 8-bit character.

As this problem occurs in most situations where data is to be transferred or stored, some designers have produced variable word-length computers.

*Variable Word-length Computers.* In these computers a word is one or more whole locations in length, and is set to the required length at each data transfer. Instructions may be of fixed or variable length but are more commonly of variable length in these machines. By setting the

word length correctly at each transfer any whole instruction can be transferred in a single operation.

There are two types of variable word-length machine:

a.    Byte machines in which each location is 8 bits long.

b. Character machines in which each location is 6 bits long. Instruction storage. In both fixed-length and variable-length machines, locations may be used to store instructions instead of data, and the stored codes are indistinguishable. In principle there is nothing to prevent a stored instruction from being used as data or vice versa! In practice storage must be used carefully to avoid such errors.

*Variation with machine size.* The smaller computers tend to be fixed word-length machines. The larger microcomputers and small minicomputers tend to have word lengths of 16 bits and sometimes 32 bits. Larger minicomputers and some mainframes have word lengths of 32 bits, and so on.

*Parity check.* We have been referring to storage as consisting of, e.g., 8, 16 or 32 bit groups. In fact each one of these groups may have a parity bit added to it in the same way that transmitted codes do. The parity bit aids the detection of errors caused through bits being "lost" in the process of storing or transmitting the data.

Note. Small computers do not always incorporate parity-checking memory systems and this can bring their reliability into question in some situations.

Physical features of main storage

To have a sound understanding of main storage we need to be aware of some features of its construction and technology. These features are illustrated here by reference to the most common type of main storage called semi-conductor memory.

STORAGE                    CELLS
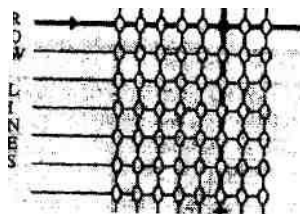


Cells in this row are "open"

COLUMN LINES The referenced cell

Fig 1. Cell Operation in Semi-conductor Memory.

*Semi-conductor memory.* The common type RAM (Random Access Memory) is described here.

a. Each bit of semi-conductor memory is represented by a single cell, which maybe regarded as a microscopic electronic circuit with two

distinguishable states used to represent "0" and "1". Cells arranged into rows and columns are built onto a single "chip" of silicon some 3mm X 3mm, in what is clearly a most advanced manufacturing process, since 2,048,000 cells may be formed on a single chip and many times more on the larger chips.

b. Each cell can be *electrically charged* or *not electrically charged* representing"1" and"0" respectively. If a row line is made electrically active it causes all cells in that row to switch open so that they can be charged or discharged through a column charge into the cell (State "1") or to allow the cell to discharge (State "0") (Fig l.). Since charge can only flow from a charged cell, an observable discharge reveals the state of a cell and provides the means of reading from memory. A cell discharged by reading from it must be recharged to preserve its state.

c Charge slowly leaks from the cells and has to be topped up constantly. This is called "refreshing". Strictly speaking this is a feature of one type of RAM called dynamic RAM. Refreshing is not required in the other type of RAM, which is called Static RAM. In static RAM the "cells" stay "latched" in a "1" state or "0"state by the provision of a constant power supply. This dependence on a constant power supply by both static and dynamic RAM means RAM memory is *volatile* (i.e., data is lost if the power supply is removed).

*ROMS.* A non-volatile semi-conductor alternative to the RAM is the ROM (Read Only Memory) in which all cell states are set permanently during manufacture. A set of instructions stored in this way is said to be "hard wired", and is often called firmware. This technique is often used to extend the instruction set of the computer by storing special-purpose subroutines. These are called micro programs. ROM is also frequently used to store data or instructions which the computer needs all the time from the moment it is switched on.

*Other types of ROM.* Apart from the basic types of ROM just described, there are several variations, which are briefly described here:

a. PROM (Programmable Read Only Memory). In contrast to normal ROM, in which the pattern of stored data is permanently incorporated into the chip using a "Mask" during the last stage of manufacture, the PROM can be programmed by the user. However, the data is held permanently once the PROM has been programmed.

b. EPROM (Erasable Programmable Read Only Memory). This is like PROM in that it may be programmed by the user but has the advantage

that it may be erased and reprogrammed. The EPROM must be removed from the computer in order to be erased. Special devices called "EPROM erasers" and "EPROM programmers" are used for these tasks.

c. EAROM (Electrically Alterable Read Only Memory). These memories can be read, erased and rewritten without removing them from the computer. However, the erasing and rewriting process is very slow compared with reading, which limits their use. For this reason they are sometimes called RMMs (Read Mostly Memories).

d.      EEROM (Electrically Erasable Read Only Memory). Essentially the same as EAROM.

Memory chips are normally assembled onto electronic boards (printed circuit boards (PCBs)). They may either be assembled onto the motherboard, which is a main PCB also occupied by the processor and other main electronic components, or they may be assembled onto separate memory boards. These memory boards are slotted into electrical connectors inside the computer. Sometimes spare "slots" are provided during manufacture so that more memory can be added later. A common type of memory assembly is the SIMM (Single Inline Memory Module), which is widely used on small computer systems.

Semi-conductor fabrication technologies

The chips used for main storage and the chips used for processors are normally made from one of two basic types of manufacturing process. The two basic fabrication processes are:

a.      BIPOLAR Technology.

a.      MOS Technology.

There are several variations on these basic technologies, e.g., Schottky bipolar and I$^2$L (Integrated Injection Logic) are variations on bipolar technology and PMOS, NMOS, CMOS, SOS, VMOS and HMOS are all variations on MOS technology. The latest Intel Pentium processors are made using BiCMOS technology. Do not attempt to remember all these variations; they are mentioned here merely to show the wide variety of technologies that exist. Indeed, the details of this technology are largely outside the scope of computer science and are more the concern of electronic engineering or computer engineering. It is the consequence of using a particular technology which is the concern of Computer Science. These types of fabrication technology have a direct bearing on the operational features of the semi-conductor devices. Factors affected include

a.  Cost.

b.  Power consumption,

c   Speed of access.

d.  Reliability.

Very broadly speaking, MOS technology is more widely used on microcomputers and BIPOLAR technology is mostly used on minicomputers and mainframe computers, which should indicate some general differences between the two technologies.

Buses

Data is normally transferred between main storage and the processor along a device called a bus which is effectively a means of sending multiple bits of data in parallel. For example, all the bits of a word may be transferred in parallel in a single operation.

On close inspection the bus is constructed from a number of lines. It is the data lines which carry the data bits. For example, a 16-bit computer might have a bus connecting the processor to main memory with 16-data lines so that the contents of one whole 16-bit word could be transferred at a time between main memory and the processor. The *wider* the bus, that is to say the more bits it can transfer in parallel, the faster the rate of data transfer may be. The bus width typically matches the word size with, for example, an 8-bit bus being used with a computer having 8-bit memory. In general, therefore, the combination of word-size and bus width is one of the significant factors influencing the speed of a computer. There are many types of bus.

Cache memory

In recent years it has become common practice to use memory chips with extra high access speeds to supplement those in main memory. The chips tend to be too expensive to use for the whole of main memory. These memory chips are used to provide a "cache", which is a temporary holding area for data which is currently subject to repeated access. The processor can access this data more quickly and there is therefore an overall speed improvement. The cache effectively sits between main storage and the processor acting as a holding area through which all data and instructions pass . The cache can also be constructed as part of the processor chip as in the case of the Intel Pentium processor some versions of which have a cache of 8,196 bytes. Old data in the cache is overwritten by new because the cache has limited capacity. To make best use of the cache an appropriate strategy must be adopted to decide what data to overwrite. Common methods are the **LFU** (Least

Frequently Used) and LRU (Least Recently Used) rules for determining which data items to overwrite.

**Comprehension check**

**Answer the questions**

1. What are the basic operational features of Main Storage?
2. Enumerate the most popular approaches to computer design.
3. Characterize two types of semi-conductor memory and their physical qualities.
4. What device is used to transfer data between main storage and processor?
5. What does a "temporary holding area" consist of? How can it help to improve the speed of a computer?

**Find the English equivalents of the following terms:**

Запоминающая ячейка; извлекать хранимую информацию; технология изготовления; адрес ячейки памяти; выборка; энергозависимость; элемент с двумя состояниями; контроль по четности; заряженная ячейка памяти; соответственно; регенерация; встроенные," зашитые" программы; шина; при ближайшем рассмотрении; перманентно встроенный.

**Translate the sentences into English, using the vocabulary of the passage:**

1. ОЗУ - энергозависимая память; при отключении напряжения питания информация, хранящаяся в ней, теряется.
2. Запись и считывание информации осуществляются подачей электрических импульсов по тем шинам матрицы, которые соединены с элементами, принадлежащими выбранной ячейки памяти.
3. Каждая ячейка памяти имеет свой уникальный адрес.
4. Потеря данных произошла, когда он случайно нажал не ту кнопку.
5. Для решения этой сложной задачи необходимо разбить ее на более мелкие.
6. Современные компьютерные технологии имеют тенденцию к очень быстрому развитию.
7. Чтобы избежать ошибок в программировании, вы должны быть очень внимательны.
8. Полное понимание вопроса приходит после его тщательного изучения.
9. Печатная плата является одним из основных элементов любого

41

компьютера.

10. Сборка данного устройства проходила в условиях строгой секретности.

Write a summary of the passage using the following vocabulary:

- Main storage
- Short-term memory
- To locate
- Data transfer
- High access speed
- Limited capacity
- Indistinguishable
- Bus width
- To top up
- To switch open
- Cache memory

Part 2
SUPPLEMENTARY READING
SOCIETY

## The underbelly of cyberspace

Depending on whom you ask, hackers are harmless pranksters, curious techies, noble freedom fighters or dangerously unpredictable cyber terrorists. As the world goes online, hackers' potential influence — and menace — grows.

Computer security expert Ray Weadock admits even he can't stop every hacker. They once seized his personal Internet account and sent obscene e-mail in his name to the White House. Another hacker trashed computer files at the Tampa school attended by Weadock's son, doing $120,000 in damage.

Cyberspace is like the new frontier, says Weadock, who heads the Tampa network security company Fortress Technologies. "There are few sheriffs out there."

With the global boom in the Internet and ever-cheaper personal computers, hacking is spreading like online kudzu. Hacking is getting more sophisticated and, in many cases, a lot nastier. And it is chipping away at the ability of the government, the military, and the business community to protect proprietary information and preserve individual privacy. Here are but a few of hundreds of recent examples:

• A Boston group of hackers known as LOpht came to Capitol Hill late last month to push Congress for tougher security measures. Computer security is so lax that they figure they could cut the entire nation off from the Internet in less than 30 minutes. One hacker, "Mudge," testified that the group could keep the global network disabled for so long, "it would definitely take a few days for people to figure out what was going on."

• A Swedish hacker last year jammed the 911 emergency phone system throughout west-central Florida, including Pasco, Hernando and Citrus counties. While the hacker was fined a mere $350 for harassment, FBI director Louis Freeh took a more jaundiced view. He called the 911 incident a "dress rehearsal for a national disaster."

• Hacker Vladimir Levin in St. Petersburg, Russia, used his laptop computer to transfer illegally at least $3.7-million from New York's Citibank to accounts around the world. Levin was later arrested at London's Heathrow Airport. In February, a U.S. judge sentenced him to

three years in prison and ordered him to pay Citibank $240,000 in restitution.

• Former University of Pittsburgh student John Vranesevich and his http://www.antionline.com Web site first publicized the news this spring that teenage hackers "Makaveli" in California and "Analyzer" in Israel had cracked the Pentagon's computer system and taken software used to track military satellites. The Pentagon later acknowledged the successful break-in.

"The ante has been upped," said Richard Power, senior analyst at San Francisco's Computer Security Institute and the author of the state-of-hacking report *Current and Future Danger.* "It's unbelievably naive to think there will not be serious criminal involvement in cyberspace in the near future."

Even the veracity of children's report cards is now suspect. One worried top administrator of a California school district told Tampa's Weadock that he could not be sure his students had the right grades. "That concern could be replicated through every school district in the United States," Weadock said.

Once an odd domain shared by computer scientists, amateur technology buffs and antisocial teenagers, the hacker world is going mainstream. Hackers can attend their own established annual conventions like New York's Beyond HOPE (Hackers On Planet Earth) or Las Vegas' DefCon to help them stay up on hacker culture and learn new hacking techniques. Next month's DefCon6 convention in Las Vegas, for example, is hosted by Seattle hacker Jeff Moss (aka "Dark Tangent"), sponsored by the likes of Jolt Cola and will draw more than 1,500 attendees.

Hackers are even catching the eye of Madison Avenue. During this year's Super Bowl ~ the world's most-expensive TV advertising venue — a computer security firm ran the first-ever ad promoting security protection against the threat of hacking.

Many hackers are benign — just intensely curious how software or computer networks work. Some hackers seem threatening but are little more than pranksters spreading online graffiti on Web sites. But a growing number are hacking for personal profit, political cause or simply to inflict damage. Many hackers, trying to distance themselves, call these online abusers "crackers."

"If we aren't vigilant, cybercrime will turn the Internet into the Wild West of the 21st century," said U.S. Attorney General Janet Reno.

Attacks on the increase

So far, individuals using their home PCs are rarely the target of hackers. But that is not the case with businesses and their employees. A study released this spring by the Computer Security Institute and the FBI's International Crime Squad found that nearly two-thirds of more than 500 organizations reported a computer security breach within the past 12 months, up from 48 percent a year ago and 22 percent the year before that.

Many hacker attacks go unreported because companies want to avoid negative publicity. Other companies stung by hackers feel compelled to tell what happened. In January, Boeing Co. advised its workers that the code used to assign temporary PINs—personal identification numbers — for their 401(k) savings accounts had been cracked, possibly by a company employee.

On a broader scale, the federal government is starting to take the threat of online mayhem to heart. "Cyberterrorism," "information warfare" and "economic espionage"— terms that did not exist until recently — are cropping up often in national security debates.

Last year, the White House created the President's Commission on Critical Infrastructure Protection. Its job is to improve the nation's defenses against online assault. It is "only a matter of time" before critical U.S. computer systems that control the nation's power grid or air traffic control networks face major attack, says commission leader and retired four-star Air Force general Robert Marsh. In February, Reno unveiled the National Infrastructure Protection Center, or NIPC. Reno said the center's mission is to protect the nation's telecommunications, technology and transportation systems. Part of that effort includes managing FBI investigations into hackers.

Some Internet watchers, like Tampa Bay security consultant Winn Schwartau, say the feds seem bent on another hacker crackdown, not unlike one in the late 1980s. But individual hackers are not the real threat, Schwartau suggests. Foreign governments and organized terrorist groups are.

"The threat is from transnational gangs," he said. "How much damage could be done to the United States online with the backing of $100-

million? A lot. And that's just chump change in the international markets."

The good and the bad

As millions of people opt to try the Internet and go online each year, new recruits join the hacker ranks every day.

That's not necessarily bad. Many are drawn to the traditional hacker culture: Anti-establishment, yes, but mostly harmless. One that embraces a passionate curiosity about computers and communications. A fervent belief that information should be free, uncensored — and shared. A strong opposition to Big Brother. Some hacker groups like LOpht, run by "Mudge" and other hackers like "SpaceRogue," "Brian Oblivion" and "Weld Pond," often find many of the "bugs" or holes in new software programs like Microsoft's xxxx

Windows NT program or Netscape's Internet browser. The group then publicizes the program's defects on the Internet. That's often how many programs get fixed. But many in the new hacker generation, when exploring a computer network, apparently ignore or never learned the hacker ethic: Look but don't touch. Consider Julio Cesar Ardita, a 23-year-old Argentine known to authorities as "Griton" (Spanish for "Screamer"), who returned voluntarily to the United States this spring, more than two years after he was first accused of hacking into university and military computer networks in the United States. Sentenced in May under a plea agreement to three years' probation and a $5,000 fine, Ardita still faces charges in his homeland.

Another big catch involved the federal sting of Carlos Felipe Salgado Jr. Known online as "Smak," Salgado was caught last year after stealing 10,000 credit card numbers off the files of an Internet service provider in California. Salgado tried to sell them for $260,000 to an undercover FBI agent.

A favorite target of hackers is America Online because its 12-million customers make a highly visible target. Fed up with a widely available program called AOL4FREE that gave users free access to the online service, AOL urged federal prosecutors to nail its creator.

The feds did. Yale University student Nicholas Ryan, known online as "Happy Hardcore," was convicted of computer fraud and sentenced last year to two years' probation, six months' home confinement and a $50 fine (after paying $62,000 in restitution). It was the first federal felony conviction of a hacker involving a private Internet online service.

AOL security chief Tatiana Gau hopes the Ryan case sent a strong message that her Internet company will not put up with hackers.

Easier said than done. Hacker groups with political agendas and an often juvenile style of protest are on the rise. In 1994, for example, when the Internet Liberation Front broke into computer networks at GE, NBC and other companies, the group denounced the companies for turning the Internet into a "cesspool of greed." Such protesters often hack prominent government or military Web sites and leave behind online graffiti. At the U.S. Justice Department Web site, for example, anti-censorship messages were left and a photo of Attorney General Reno was swapped with that of Adolf Hitler.

Hackers also altered the Central Intelligence Agency site to read "Central Stupidity Agency." College basketball's NCAA site (during 1997's finals) was doctored to display a "white power" symbol. And Valujet (now called AirTran), the airline whose passenger jet crashed in the Everglades in 1996, had its Web site altered last year by hackers who inserted an image of a burning plane and the line "Fly us because crashing is fun."

In most cases, hacked sites are quickly discovered, shut down and fixed, but not before the protest messages become part of Internet lore.

## Legendary exploits

For hacker wannabes, role models are plentiful. Some big-name hackers from the past include John Draper, who discovered that the tone of a whistle given away in a box of cereal could, when blown into a pay phone, trigger the country's phone system to allow free calls. That discovery earned Draper an international following and, in a salute to the cereal brand, his still famous hacker handle: "Captain Crunch."

Even Steve Jobs and Steve Wozniak, founders of Apple Computer, started out as hackers. And rival hacker gangs known as the Masters of Deception and the Legion of Doom dominated the mid-1980s with their successful invasions of the country's telephone networks.

In recent years, Kevin Mitnick has emerged as the most hyped name in hackerdom. Mimicks hacking exploits of the late 1980s and early '90s inspired a media frenzy and multiple books. His current jail term has made him the poster boy of hacker protests against Big Brother. "Free Kevin" protest messages still litter hacked Web sites and the hotel hallways of hacker conventions.

If such prominent hackers are not inspiring, dozens of others are making a name for themselves. How to hack isn't a well-hidden secret. Hacking guides, tips, tools and manifestoes can be found online at an estimated 2,000 Web sites and 440 bulletin boards.

And if real hackers don't inspire the up-and-comers, pop culture will. Hackers as nerdy geniuses battling Big Brother is a popular movie theme. The 1983 movie *War Games* featured actor Matthew Broderick hacking innocently into the U.S. military NORAD network to play Global Thermonuclear War (for real). The movie fired the imagination of teenagers to the thrill of computer literacy. In 1986, Broderick surfaced again in *Ferris Bueller's Day Off,* this time using his home PC to change student grades at school.

In the 1992 movie *Sneakers,* hackers led by actor Robert Redford broke into a highly secure corporation. Tapping an especially sensitive topic, hackers in 1995's *The Net* stole the identity of Sandra Bullock's character by altering her personal computer records. Now computer hacking is a common subplot in dozens of popular flicks, from *Jurassic Park* to *Mission: Impossible.*

Hacking became so hip that the 1995 movie *Hackers* tried to capture the culture of the underground computer scene. The movie's weak plot so incensed real hackers that they seized the movie's promotional Web site, defaced the online photos of the film's characters and left behind a scathing review: "no plot or creative thought." Hollywood's MGM/UA studio, ever vigilant to any kind of free publicity, kept the hacked Web site available, graffiti intact, for online viewers.

A law enforcement backlash

The hyper-growth of malevolent hacking is not going unchecked. In addition to Attorney General Reno's recent edicts, the FBI and the Secret Service are building a national network of computer-literate agents that can help monitor, track and pursue online hackers.

Even the Florida Department of Law Enforcement has trained several of its agents to specialize in computer-related crimes. Still, in contrast to the explosive growth of the Internet, the number of law enforcement officers with such training remains small.

The antics of Cornell University graduate student Robert Tappan Morris Jr. helped spur the creation of an Internet computer SWAT team. In an experiment gone haywire, Morris in 1988 let loose a computer virus known as a "worm" that replicated itself across the Internet and crashed a

tenth of the network. Morris, the son of a National Security Agency computer expert, was later convicted and fined $10,000.

In the aftermath, the Computer Emergency Response Team, or CERT, was formed at Carnegie Mellon University in Pittsburgh to investigate attacks on computer networks and, when possible, offer remedies.

CERT is very busy these days. As many as 200 new viruses with such onerous names as Antichrist, Bad Taste, Damage and HTV are identified each month.

Any high-profile body, it seems, is fair game to hackers. Just ask the police in New York City.

On April 15, 1996, callers to the NYPD heard this message: "You have reached the New York City Police Department. For any real emergencies, dial 119. Anyone else - we're a little busy right now eating some doughnuts and having coffee."

Or ask *Time* magazine writer Joshua Quittner, who co-wrote *Masters of Deception,* a 1995 book about the New York-based hacker gang. After the book was published, Quittner's e-mail service was trashed by hackers. His phone was re-routed several times, first to an out-of-state answering machine, then to a phone-sex number and once to 1-800-EAT-S—.

It took half a dozen unlisted numbers and a year of phone taps by the phone company's security folks to stop the problem.

The experience influenced Quittner's writing plans, as he explained in a *Time* article.

"Write another hacker book? I'd rather take on the Scientologists."

## A HISTORY OF HACKING

Hacking has been around for more than a century. In the 1870s, several teenagers nwere flung off the country's brand new phone system by enraged authorities. Here's a peek at how busy hackers have been in the past 35 years.

Early 1960s

University facilities with huge mainframe computers, like MITs artificial intelligence lab, become staging grounds for hackers. At first, "hacker" was a positive term for a person with a mastery of computers who could push programs beyond what they were designed to do.

Early 1970s

John Draper makes a long-distance call for free by blowing a precise tone into a telephone that tells the phone system to open a line. Draper

discovered the whistle as a give-away in a box of children's cereal. Draper, who later earns the handle "Captain Crunch," is arrested repeatedly for phone tampering throughout the 1970s.

Yippie social movement starts *YIPUTAP* (Youth International Party Line/Technical Assistance Program) magazine to help phone hackers (called "phreaks") make free long-distance calls.

Two members of California's Homebrew Computer Club begin making "blue boxes,"devices used to hack into the phone system. The members, who adopt handles "Berkeley Blue" (Steve Jobs) and "Oak Toebark" (Steve Wozniak), later go on to found Apple Computer.

Early 1980s

Author William Gibson coins the term "cyberspace" in a science fiction novel called *Neuromancer.*

In one of the first arrests of hackers, the FBI busts the Milwaukee-based 414s (named after the local area code) after members are accused of 60 computer break-ins ranging from Memorial Sloan-Kettering Cancer Center to Los Alamos National Laboratory.

Comprehensive Crime Control Act gives Secret Service jurisdiction over credit card and computer fraud.

Two hacker groups form, the Legion of Doom in the United States and the Chaos Computer Club in Germany.

*2600: The Hacker Quarterly* is founded to share tips on phone and computer hacking.

Late 1980s

The Computer Fraud and Abuse Act gives more clout to federal authorities.

Computer Emergency Response Team is formed by U.S. defense agencies. Based at Carnegie Mellon University in Pittsburgh, its mission is to investigate the growing volume of attacks on computer networks.

At 25, veteran hacker Kevin Mitnick secretly monitors the e-mail of MCI and Digital Equipment security officials. He is convicted of damaging computers and stealing software and is sentenced to one year in prison.

First National Bank of Chicago is the victim of a $70-million computer heist.

An Indiana hacker known as "Fry Guy" — so named for hacking McDonald's — is raided by law enforcement. A similar sweep occurs in

Atlanta for Legion of Doom hackers known by the handles "Prophet," "Leftist" and "Urvile."

Early 1990s

After AT&T long-distance service crashes on Martin Luther King Jr. Day, law enforcement starts a national crackdown on hackers. The feds nab St. Louis' "Knight Lightning" and in New York grab Masters of Deception trio "Phiber Optik," " Acid Phreak" and "Scorpion." Fellow hacker "Eric Bloodaxe" is picked up in Austin, Texas.

Operation Sundevil, a special team of Secret Service agents and members of Arizona's organized crime unit, conducts raids in 12 major cities, including Miami. A 17-month search ends in the capture of hacker Kevin Lee Poulsen ("Dark Dante"), who is indicted for stealing military documents.

Hackers break into Griffith Air Force Base, then pewwwte computers at NASA and the Korean Atomic Research Institute. Scotland Yard nabs "Data Stream," a 16-year-old British teenager who curls up in the fetal position when seized. A Texas A&M professor receives death threats after a hacker logs on to his computer from off-campus and sends 20,000 racist e-mail messages using his Internet address. In a highly publicized case, Kevin Mitnick is arrested (again), this time in Raleigh, N.C., after he is tracked down via computer by Tsutomu Shimomura at the San Diego Supercomputer Center.

Late 1990s

Hackers break into and deface federal Web sites, including the U.S. Department of Justice, U.S. Air Force, CIA, NASA and others.

Report by the General Accounting Office finds Defense Department computers sustained 250,000 attacks by hackers in 1995 alone.

A Canadian hacker group called the Brotherhood, angry at hackers being falsely accused of electronically stalking a Canadian family, break into the Canadian Broadcasting Corp. Web site and leave message: "The media are liars." Family's own 15-year-old son eventually is identified as stalking culprit.

Hackers pierce security in Microsoft's NT operating system to illustrate its weaknesses.

Popular Internet search engine Yahoo! is hit by hackers claiming a "logic bomb" will go off in the PCs of Yahool's users on Christmas Day 1997 unless Kevin Mitnick is released from prison. "There is no virus," Yahoo! spokeswoman Diane Hunt said.

# SECOND SIGHT

The plastic cards in our wallets date back to the 50s, their magnetic stripes to the 60s and their smart chips to the 70s. They really are getting antiquated, which is why -despite their dominance - there's a feeling there might be a better way of paying for things online. Yet in that environment, it's the business model, not the technology that might be worth re-examining.

It wouldn4 make sense to get your shopping at Tesco, take the bill to Barclays, pay it, take the receipt from Barclays back to Tesco and then pick up the shopping, as shoppers at Soviet department stores used to do. It would, however, be much cheaper for Tesco and Barclays if you did: Tesco wouldn't need cash drawers or point-of-sale (POS) devices to accept payment cards and Barclays wouldn't need to issue cards or process the deposits. So, instead, you give your bank-issued payment card to the shop, which passes on the details to another bank, which discusses it with your bank and then gets back to the shop and says it's OK.

Instead, when we take our basket to the checkout, we bring together two business processes that ought to be distinct: executing transactions and financing them. Why does the shop handle your cash, card or cheque when these clearly belong to the world of banking? The answer is that it would be crazy to do it any other way. In the physical world.

Similarly, when credit cards were introduced, it made no sense for every bank to run a cable to every shop in the world: communications were scarce and expensive. Hence MasterCard and Visa. When you present your card in Tesco, Tesco doesn4 care which bank issued it. Tesco passes the details to its own bank, which passes them via the credit card networks to your bank.

Today, communications are not scarce or expensive, and on the internet, every shop is connected to every bank. Yet we still use the old payment paradigm: just because we used to pay for things in a particular way, we've taken that model into the new environment. But we don't have to pay for things that way: we can make the process on the internet better than the process in Tesco.

Suppose that when you bought something from Amazon, instead of giving Amazon your payment card details, Amazon sends you a bill. You then present the bill to your bank and agree payment. Your bank gives you a receipt in return. You pass the receipt to Amazon: Amazon's bank

can easily check that the receipt has been paid (using digital signature technology) and it sends your book.

This isn't a new idea. In the early days of internet commerce, quite a few people realised that the payment process ought to be the other way round. There were a number of efforts to introduce this kind of payment scheme. They have failed, so far, to reach the mass market but, as internet shopping continues to grow, perhaps it's time to think again about a more Soviet-style approach to baskets and checkouts.

## INTERNET ADDICTION GUIDE

### What is Internet Addiction Disorder (IAD)?

What "Internet addiction disorder" (IAD) is still difficult to define at this time. Much of the original research was based upon the weakest type of research methodology, namely exploratory surveys with no clear hypothesis or rationale backing them. Coming from an atheoretical approach has some benefits, but also is not typically recognized as being a strong way to approach a new disorder. More recent research has expanded upon the original surveys and anecdotal case study reports. However, as I will illustrate below later, even these studies don't support the conclusions the authors claim.

The original research into this disorder began with exploratory surveys, which cannot establish *causal* relationships between specific behaviors and their cause. While surveys can help establish descriptions of how people feel about themselves and their behaviors, they cannot draw conclusions about whether a specific technology, such as the Internet, has actually *caused* those behaviors. Those conclusions that are drawn are purely speculative and subjective made by the researchers themselves. Researchers have a name for this logical fallacy, ignoring a common cause. It's one of the oldest fallacies in science, and one still regularly perpetrated in psychological research today.

Do some people have problems with spending too much time online? Sure they do. Some people also spend too much time reading, watching television, and working, and ignore family, friendships, and social activities. But do we have *TV addiction disorder, book addiction, and work addiction* being suggested as legitimate mental disorders in the same category as schizophrenia and depression? I think not. It's the tendency of some mental health professionals and researchers to want to label everything they see as potentially harmful with a new

diagnostic category. Unfortunately, this causes more harm than it helps people.

What most people online who think they are addicted are probably suffering from is the desire to not want to deal with other problems in their lives. Those problems may be a mental disorder (depression, anxiety, etc.), a serious health problem or disability, or a relationship problem. It is no different than turning on the TV so you won't have to talk to your spouse, or going "out with the boys" for a few drinks so you don't have to spend time at home. Nothing is different except the modality. What some very few people who spend time online without any other problems present *may* suffer from is *compulsive* over-use. Compulsive behaviors, however, are already covered by existing diagnostic categories and treatment would be similar. It's not the technology (whether it be the Internet, a book, the telephone, or the television) that is important or addicting — it's the behavior. And behaviors are easily treatable by traditional cognitive-behavior techniques in psychotherapy. Case studies, the alternative to surveys used for many conclusions drawn about online overuse, are just as problematic. How can we really draw any reasonable conclusions about *millions of people online* based upon one or two case studies? Yet media stories, and some researchers, covering this issue usually use a case study to help "illustrate" the problem. All a case study does is influence our emotional reactions to the issue; it does nothing to help us further understand the actual problem and the many potential explanations for it. Case studies on an issue like this are usually a red flag that help frame the issue in an emotional light, leaving hard, scientific data out of the picture. It is a common diversionary tactic.

There is more research that needs to be critically examined here, which I will provide descriptive analyses of shortly.

Why Does the Research Leave Something to Be Desired?

Well, the obvious answer is that many of the original researchers into the phenomenon known as IAD were actually clinicians who decided to conduct a survey. Usually doctoral training is sufficient to create and test a survey, yet the psychometric properties of these surveys are never released. (Perhaps because they were never conducted in the first place? We simply do not know.) The obvious confounds are never controlled for in most of these surveys. Questions about pre-existing or a history of mental disorders (e.g., depression, anxiety), health problems or

disabilities, or relationship problems are absent from these surveys. Since this is one of the most obvious alternative explanations for some of the data being obtained (for example, see Storm King's article, Is the Internet Addictive, or Are Addicts Using the Internet? below), it is very surprising these questions are left off. It taints all the data and make the data virtually useless.

Other factors are simply not controlled for. The current Internet population is nearly 50/50 in terms of proportion of men to women. Yet people are still drawing conclusions about this same group of people based upon survey samples that have 70-80% men, comprised mostly of white Americans. Researchers barely mention these discrepancies, all of which will again skew the results.

Research done in a particular area should also agree about certain very basic things after a time. Years have gone by and there are more than a few studies out there looking at Internet addiction. Yet none of them agree on a single definition for this problem, and all of them vary widely in their reported results of how much time an "addict" spends online. If they can't even get these basics down, it is not surprising the research quality still suffers.

More research has been done since the original surveys were released in 1996. This newer research has been conducted by more independent researchers with clearer hypotheses and stronger, less biased population sets. More about these studies will be discussed in updates to this article.

Do You Spend Too Much Time Online?

In relation to what or whom? Time alone cannot be an indicator of being *addicted* or engaging in compulsive behavior. Time must be taken in context with other factors, such as whether you're a college student (who, as a whole, proportionally spend a greater amount of time online), whether it's a part of your job, whether you have any pre-existing conditions (such as another mental disorder; a person with depression is more likely to spend more time online than someone who doesn't, for instance, often in a virtual support group environment), whether you have problems or issues in your life which may be causing you to spend more time online (e.g., using it to "get away" from life's problems, a bad marriage, difficult social relations), etc. So talking about whether you spend *too much* time online without this important context is useless.

What Makes the Internet So Addictive?

Well, as I have shown above, the research is exploratory at this time, so suppositions such as what makes the Internet so "addictive" are no better than guesses. Since other researchers online have made their guesses known, here are mine. Since the aspects of the Internet where people are spending the greatest amount of time online have to do with social interactions, it would appear that socialization is what makes the Internet so "addicting." That's right — plain old hanging out with other people and talking with them. Whether it's via e-mail, a discussion forum, chat, or a game online (such as a MUD), people are spending this time exchanging information, support, and chit-chat with other people like themselves. Would we ever characterize any time spent in the real world with friends as "addicting?" Of course not. Teenagers talk on the phone for hours on end, with people they see everyday! Do we say they are addicted to the telephone? Of course not. People lose hours at a time, immersed in a book, ignoring friends and family, and often not even picking up the phone when it rings. Do we say they are addicted to the book? Of course not. If some clinicians and researchers are now going to start defining addiction as social interactions, then every real-world social relationship I have is an addictive one.

Socializing — talking — is a very "addictive" behavior, if one applies the same criteria to it as researchers looking at Internet addiction do. Does the fact that we're now socializing with the help of some technology (can you say, "telephone"?) change the basic process of socialization? Perhaps, a bit. But not so significantly as to warrant a disorder. Checking e-mail, as Greenfield claims, is not the same as pulling a slot-machine's handle. One is social seeking behavior, the other is reward seeking behavior. They are two very different things, as any behaviorist will tell you. It's too bad the researchers can't make this differentiation, because it shows a significant lack of understanding of basic behavioral theory.

## Alternative Hypotheses

In addition to those previously discussed, here is an alternative hypothesis that no research to date has seriously considered — that the behaviors we are observing are phasic. That is, for most people with "Internet addiction," they are likely newcomers to the Internet. They are going through the first stage of acclimating themselves to a new environment ~ by fully immersing themselves in it. Since this environment is so much larger than anything we've ever seen before, some people get "stuck" in the acclimation (or enchantment) stage for a

longer period of time than is typical for acclimating to new technologies, products, or services. Walther (1999) made a similar observation based upon the work of Roberts, Smith, and Pollack (1996). The Roberts et al. study found that online chat activity was phasic — people first were enchanted by the activity (characterized by some as obsession), followed by disillusionment with chatting and a decline in usage, and then a balance was reached where the level of chat activity normalized.

For existing online users, my model allows for overuse as well, since the overuse is defined by finding a new online activity. I would argue, though, that existing users have a much more easier time successfully navigating through these stages for new activities they find online than newcomers to the Internet. It is possible, however, for an existing user to find a new activity (such as an attractive chat room or newsgroup or Website) that could lead them back into this model.

Note one important distinction about my model... It makes the assumption that since all online activity is phasic to some degree, all people will eventually get to Stage III on their own. Just like a teenager learns to not spend hours on the telephone every night on their own (eventually!), most adults online will also learn how to responsibly integrate the Internet into their lives. For some, this integration simply takes longer than others.

<u>What Do I Do If I Think I Have It?</u>
First, don't panic. Second, just because there is a debate about the validity of this diagnostic category amongst professionals doesn't mean there isn't help for it. In fact, as I mentioned earlier, help is readily available for this problem without needing to create all this hoopla about a new diagnosis.

If you have a life problem, or are grappling with a disorder such as depression, seek professional treatment for it. Once you admit and address the problem, other pieces of your life will fall back into place.

Psychologists have studied compulsive behaviors and their treatments for years now, and nearly any well-trained mental health professional will be able to help you learn to slowly curve the time spent online, and address the problems or concerns in your life that may have contributed to your online overuse, or were caused by it. No need for a specialist or an online support group.

# HUMOR

# Funny stories about Microsoft

Genesis According To Microsoft

In the beginning, there was nothing but Apple. And the PC was without form and void, and the darkness was on the face of its hard drive. And Bill said, Let there be DOS: And there was DOS. And Bill looked upon it, and it was good, and with it the PC slew the Apple. And DOS grew and grew, until its number was legion if you counted the decimal points, and still it was good.

And Bill grew large with ambition, and he decreed there should be a processor of words; and lo, there was Word. And Bill sayeth, Let there be a thingy for the crunching of numbers, and lo, there was Excel, and did his kingdom grow apace.

But there had arisen in the land the thing called Macintosh, sprung from the intransigent Apple-men, and Bill looked upon it, and it was better Rapidly did he decree that Word should be made to run upon it, and after that Excel, and then all the other fruits of his efforts, but still he was wrathful.

So Bill did order his minions to come forth with Windows, and when they did, he looked upon it, and it was bad.

So he sent them back to try again, assuring all the world they would get it right this time, yet they did not.

Unrelenting, Bill forced yet a third mighty blow, and when it came forth, Bill did order his trumpets to blow, and his chorus to sing, and his criers to cry, until the din could be heard throughout the land; and when he looked upon this third version of Windows, he saw it was not all that great, but like hotcakes did it sell.

And thus did Bill gloat, for the world proclaimed he had matched the lowly Macintosh, and his praises were sung throughout the land.

And so he ordered another, mightier, more magnificient version made, and his henchmen and henchwomen did labor hard.

Still it was not forthcoming in the year promised, nor the year promised next, and rumors did abound, and magazines did overflow with secret peeks, and columnists did heap their scorn upon it. And came the minions of the Justice Department, bent upon proving Bill monopolous, yet before his wrath did they quail, and proclaim him innocent, mostly.

And that which was once called Chicago became known as Windows 95, and the suspense built throughout the land, and Bill, remembering what had gone before, set about building a great Hype.

Into his Hype he put the greatest mouths of the land, and scattered the fruits of his profits so heavily that he bought hosts of angels to sing, and Rolling Stones songs, and trumpets and horns and drums without number. As the time of birthing grew nigh, he purchased television time without end, and appeared thereon himself, and bought entire editions of newspapers to give away unto the faithful, and traveling circuses to visit each great city.

And so when Windows 95 was born did hysteria rule the land, as the choirs sang and the trumpets and horns did blare and the televisions and the newspapers charge their followers to go forth and buy.

Heeding this, the populace did rush to the marketplace at the stroke of midnight, when even the cock doth sleep, and did push and shove and come even to blows the better to secure their own copies lest they be thought ignorant, or uncool, or hamsters in the eyes of Bill.

And Bill looked upon what he had wrought, and he giggled, and rabbeth his hands together, and even in the moment of his triumph, began to think of Next Time.


Microsoft Bids to Acquire Catholic Church

VATICAN CITY (AP)-In a joint press conference in St. Peter's Square this morning, MICROSOFT Corp. and the Vatican announced that the Redmond software giant will acquire the Roman Catholic Church in exchange for an unspecified number of shares of MICROSOFT common stock. If the deal goes through, it will be the first time a computer software company has acquired a major world religion.

With the acquisition, Pope John Paul II will become the senior vice-president of the combined company's new Religious Software Division, while MICROSOFT senior vice-presidents Michael Maples and Steven Ballmer will be invested in the College of Cardinals, said MICROSOFT Chairman Bill Gates.

"We expect a lot of growth in the religious market in the next five to ten years," said Gates. "The combined resources of MICROSOFT and the Catholic Church will allow us to make religion easier and more fun for a broader range of people."

Through the MICROSOFT Network, the company's new on-line service, "we will make the sacraments available on-line for the first time" and revive the popular pre-Counter-Reformation practice of selling indulgences, said Gates. "You can get Communion, confess your

sins, receive absolution—even reduce your time in Purgatory~all without leaving your home."

A new software application, MICROSOFT Church, will include a macro language which you can program to download heavenly graces automatically while you are away from your computer.

An estimated 17,000 people attended the announcement in St Peter's Square, watching on a 60-foot screen as comedian Don Novello—in character as Father Guido Sarducci-hosted the event, which was broadcast by satellite to 700 sites worldwide.

Pope John Paul II said little during the announcement. When Novello chided Gates, "Now I guess you get to wear one of these pointy hats," the crowd roared, but the pontiffs smile seemed strained.

The deal grants MICROSOFT exclusive electronic rights to the Bible and the Vatican's prized art collection, which includes works by such masters as Michelangelo and Da Vinci. But critics say MICROSOFT will face stiff challenges if it attempts to limit competitors' access to these key intellectual properties.

"The Jewish people invented the look and feel of the holy scriptures," said Rabbi David Gottschalk of Philadelphia. "You take the parting of the Red Sea-we had that thousands of years before the Catholics came on the scene."

But others argue that the Catholic and Jewish faiths both draw on a common Abrahamic heritage. "The Catholic Church has just been more successful in marketing it to a larger audience," notes Notre Dame theologian Father Kenneth Madigan. Over the last 2,000 years, the Catholic Church's market share has increased dramatically, while Judaism, which was the first to offer many of the concepts now touted by Christianity, lags behind.

Historically, the Church has a reputation as an aggressive competitor, leading crusades to pressure people to upgrade to Catholicism, and entering into exclusive licensing arrangements in various kingdoms whereby all subjects were instilled with Catholicism, whether or not they planned to use it. Today Christianity is available from several denominations, but the Catholic version is still the most widely used.

The Church's mission is to reach "the four corners of the earth," echoing MICROSOFT'S vision of "a computer on every desktop and in every home." Gates described MICROSOFT'S long-term strategy to develop a scalable religious architecture that will support all religions through

emulation. A single core religion will be offered with a choice of interfaces according to the religion desired—"One religion, a couple of different implementations," said Gates.

The MICROSOFT move could spark a wave of mergers and acquisitions, according to Herb Peters, a spokesman for the U.S. Southern Baptist Conference, as other churches scramble to strengthen their position in the increasingly competitive religious market.

If Restaurants Worked Like Microsoft Patron:
 Server!
 Server: Hi, my name is Bill, and I'll be your Support
 Server. What seems to the problem?

 Patron: There's a fly in my soup!
 Server: Try again; maybe the fly won't be there this time.
 Patron: No, it's still there.
 Server: Maybe it's the way you're using the soup; try eating it with a fork
 instead.
Patron: Even when I use the fork, the fly is still there. Server: Maybe the soup is incompatible with the bowl; what kind of bowl are you using?

 Patron: A SOUP bowl!
 Server: Hmmm, that should work. Maybe it's a configuration problem; how was the bowl set up?
Patron: You brought it to me on a saucer; what has that got to do with the fly in my soup?!
Server: Can you remember everything you did before you noticed the fly in your soup?
Patron: I sat down and ordered the Soup of the Day! Server: Have you considered upgrading to the latest Soup of the Day?
Patron: You have more than one Soup of the Day each day??

 Server: Yes, the Soup of the Day is changed every hour.

 Patron: Well, what is the Soup of the Day now?
 Server: The current Soup of the Day is tomato.
 Patron: Fine. Bring me the tomato soup and the check. I'm running late now.
[Server leaves and returns with another bowl of soup and the check.]

Server: Here you are, Sir. The soup and your check.
Patron: This is potato soup.

Server: Yes, the tomato soup wasn't ready yet.
Patron: Well, I'm so hungry now, I'll eat anything.
[Server leaves.]
Patron: Server! There's a gnat in my soup!
The check:

Soup of the Day     $5.00
Upgrade to newer Soup of the Day   $2.50
Access to support   $ 1.00

Funny stories **about Windows**

Idiot's Guide to Windows 95 Ads
•     Multitasking: You can crash several programs all at once. No waiting!
•     Built-in Networking: You can crash several PC's all at once. No need to buy Novell Personal Netware or LANtastic to crash.
•     Microsoft Network: Connect with other Windows 95 users and talk about your crash experiences. Support groups in different cities will be organized.
•     PnP: Plug and Pray (that it works)
•     Multimedia: Experience the immense sight and sound of crashing.
.     Compatible with existing software: It will also crash your existing software.
•     Increased Productivity: You will need to increase your budget to buy more

products like RAM and HardDnves. Better yet, get a new computer! That's product ivity.
.     User-Friendly: Picture of clouds
.     State of the Art: Pay for Bill's next bid for a work of art.
•     Macintosh-like: It took Microsoft eleven years and it's not even original.
. Online Registration: Dial into Microsoft and let them snoop around your hard drive. This will guarantee you a place in Microsoft's files for the rest of your life.
    MS Plus: More money for Bill's plus side.

• Optimize: It will increase the utilization of your hard drive and cpu so much so that you'll end up upgrading your system. See "Increased Productivity".

Is Windows a Virus?

McAfee-Question : Is Windows a virus ?

No, Windows is not a virus. Here's what viruses do:

They replicate quickly - okay, Windows does that.

Viruses use up valuable system resources, slowing down the system as they do so - okay, Windows does that.

Viruses will, from time to time, trash your hard disk - okay, Windows does that too.

Viruses are usually carried, unknown to the user, along with valuable programs and systems. Sigh... Windows does that, too.

Viruses will occasionally make the user suspect their system is too slow (see 2) and the user will buy new hardware. Yup, that's with Windows, too.

Until now it seems Windows is a virus but there are fundamental differences: Viruses are well supported by their authors, are running on most systems, their program code is fast, compact and efficient and they tend to become more sophisticated as they mature.

So Windows is not a virus. It's a bug.

Windows for Telepaths

The Usenet Oracle has pondered your question deeply.

Your question was:

Oh wonderful and wise Oracle tell me how would one implement a version of Windows TP    Windows for TelePaths ?

And in response, thus spake the Oracle:

You open the box labeled "Windows TP", carefully extracting the pouch labeled "License Agreement". You examine the contents of the pouch, finding an inflatable beanie bearing the Windows logo rather than the familiar 3.5" diskette package. You inflate the beanie, insert two "C"-size batteries (not included), and carefully place it on your head.

You press the Start button.

Immediately, the image of an hourglass comes to your mind. You find yourself trapped; unable to move anything in your body save your eyes.

After an indeterminable delay, you regain control of your senses. You are suddenly compelled to speak your name and business affiliation.

You then retrieve your Windows TP package and chant the Product-ID number.

Suddenly you see the words "Windows is detecting new hardware" flash before your eyes.

You crash to the floor, writhing in agony. You feel every muscle in your body contract and retract in turn. Your mind is filled with the image of a blue inchworm, creeping slowly across a grey field.

The creature finally reaches the edge of its domain, and your seizure ceases. You take a moment to regain your composure, and you are reminded of your high school anatomy course as a complete listing of every organ in your body appears before your eyes. You browse the list for a moment, and utter the phrase "OK". After a short delay, you hear the sound of a trumpet echo through the recesses of your mind.

You find yourself in a large, barren space. You look around, and discover images labeled "My Brain", "Recycle Bin, and "Set up the Microsoft Network". You feel compelled to utter me word "Start", after which a list of options floods your mind.

Weary from the detection phase, you utter the word "Shut down". You close your eyes, and blackness surrounds you. You feel yourself start to drift into sleep. Your peace is interrupted, however, as a bright orange light invades your nothingness. "It's now safe to shut down your mind".

You drift into unconsciousness, and sleep for several hours.

When you awaken, you are frozen in place as you see clouds and blue cycling colors.

After a short eternity, the familiar "My Brain" icon reappears in your mind. But something is terribly wrong; you can feel it in your gut. Just outside the range of primary vision, you can sense something lurking about you on all four sides.

You slowly look up, and see the word "Safe Mode" glaring back at you. You back away slowly, swivel your head, and there it is, behind you as well. Your heartbeat quickened and you are terrified as you turn to your left and your right and it meets you there as well, its cold, heartless glare filling your soul with despair. Quickly, you summon Control Panel, System, Device Manager. You feel yourself frantically gasping for air as you run through the list of installed devices.

You come upon "Respiratory System" and are horrified to see a black exclamation point on a yellow field next to the entry "Lungs". You close your eyes and utter the word "Properties". On the closed curtains of your eyelids, you see your life flashing before your eyes.

You force yourself to concentrate on your situation, attempting to discover which system devices are in conflict, when suddenly your entire body seizes up in pain.

You lose all sense of reality. You are floating through the clouds as you hear a voice echo through your mind: "This program has performed an illegal operation and will be terminated." You start to black out and suddenly you remember your situation.

You stare in horror at your blue extremities, knowing that, without oxygen, you will not last much longer. With all the consciousness you can muster, you force yourself...

To reboot.

You awaken in a place that is dark, but familiar. A solitary white prompt on a black field greets you. You look behind you and see the wreckage of the operating system that nearly spelled your demise. "Cannot find a file that may be needed to run Windows". You turn around to face the prompt, and a wide grin comes across your face. You take a deep breath and revel in the life-giving atmosphere. You laugh as you utter the words, "DELTREE WINDOWS".

Suddenly you find yourself on the floor of your home. You find the charred remains of the Windows TP beanie littering the floor. You carefully gather them up, stack them neatly on an altar, and bum them, promising yourself never to risk your life with Microsoft again. You bury the ashes, knowing that your life is again in order.

You owe the Oracle a copy of Windows TP and Bill Gates' home address.

Funny stories about Bill Gates

Bill Gates at Heaven's Gate

Shortly after Bill Gates was killed in a freak accident, he found himself being sized up by Saint Peter. "Bill, this is a tough call. You've made great technological advancements with Microsoft, but you've also given us Windows 95. I think I'm going to let you choose between heaven and hell."

"That sounds fair," Gates replied. "Can I have a look at hell first?"

Saint Peter showed his guest a wonderland of sunny beaches, beautiful women, sumptuous food and an ideal climate. "If this is hell," Gates explained, "I want to see heaven."

Saint Peter led the way through billowy clouds filled with angels playing golden harps. "Hmm," Gates pondered. "This is nice, but I think I prefer hell."

Two weeks later, Saint Peter went to check on the billionaire. He found him shackled to a wall, surrounded by shooting flames and tormenting demons. "Saint Peter!"

Gates cried. "This is awful! This is nothing like the hell I visited. What happened to that other place, the one with the beaches, the beautiful women and the delicious food?"

"Oh, that," Saint Peter replied. "That was just a demo."

Bill Gates Goes to Heaven

Bill Gates died and, much to everyone's surprise, went to Heaven. When he got there, he had to wait in the reception area.

Heaven's reception area was the size of Massachusetts. There were literally millions of people milling about, living in tents with nothing to do all day. Food and water were being distributed from the backs of trucks, while staffers with clipboards slowly worked their way through the crowd. Booze and drugs were being passed around.

Fights were commonplace. Sanitation conditions were appalling. All in all, the scene looked like Woodstock gone metastatic.

Bill lived in a tent for three weeks until, finally, one of the staffers approached him.

The staffer was a young man in his late teens, face scarred with acne. He was wearing a blue T-shirt with the words TEAM PETER emblazoned on it in large yellow lettering.

"Hello," said the staffer in a bored voice that could have been the voice of any clerk in any overgrown bureaucracy. "My name is Gabriel and I'll be your induction coordinator." Bill started to ask a question, but Gabriel interrupted him. "No, I'm not the Archangel Gabriel. I'm just a guy from Philadelphia named Gabriel who died in a car wreck at the age of 17. Now give me your name, last name first, unless you were Chinese in which case it's first name first."

"Gates, Bill." Gabriel started searching though the sheaf of papers on his clipboard, looking for Bill's Record of Earthly Works. "What's going on here?" asked Bill.

"Why are all these people here? Where's Saint Peter? Where are the Pearly Gates?"

Gabriel ignored the questions until he located Bill's records. Then Gabriel looked up in surprise. "It says here that you were the president of a large software company. Is that right?"

'Yes."

"Well then, do the math chip-head! When this Saint Peter business started, it was an easy gig. Only a hundred or so people died every day, and Peter could handle it all by himself, no problem. But now there are over five billion people on earth. Jesus, when God said to 'go forth and multiply,' he didn't say 'like rabbits!' With that large a population, ten thousand people die every hour. Over a quarter million people a day.

Do you think Peter can meet them all personally?"

"I guess not."

'You guess right. So Peter had to franchise the operation. Now, Peter is the CEO of Team Peter Enterprises, Inc. He just sits in the corporate headquarters and sets policy.

Franchisees like me handle the actual inductions." Gabriel looked though his paperwork some more, and then continued. "Your paperwork seems to be in order.

And with a background like yours, you'll be getting a plum job assignment."

"Job assignment?"

"Of course. Did you expect to spend the rest of eternity drinking ambrosia? Heaven is a big operation. You have to pull your weight around here!" Gabriel took out a triplicate form, had Bill sign at the bottom, and then tore out the middle copy and handed it to Bill.

"Take this down to induction center #23 and meet up with your occupational orientator. His name is Abraham -Bill started to ask a question, but Gabriel interrupted him. "No, he's not *that* Abraham."

Bill walked down a muddy trail for ten miles until he came to induction center #23.

He met with Abraham after a mere six-hour wait.

"Heaven is centuries behind in building its data processing infrastructure," explained Abraham. "As you've seen, we're still doing everything on paper. It takes us a week just to process new entries." "I had to wait *three* weeks," said Bill. Abraham stared at Bill angrily, and Bill realized that he'd made a mistake. Even in Heaven, it's best not to contradict a bureaucrat. "Well," Bill offered, "maybe that Bosnia thing has you guys backed up."

Abraham's look of anger faded to mere annoyance. "Your job will be to supervise Heaven's new data processing center. We're building the largest computing facility in creation. Haifa million computers connected by a multi-segment fiber optic network, all running into a back-end server network with a thousand CPUs on a gigabit channel. Fully fault tolerant. Fully distributed processing. The works."

Bill could barely contain his excitement. "Wow! What a great job! This is really Heaven!"

"We're just finishing construction, and we'll be starting operations soon. Would you like to go see the center now?"

'You bet!"

Abraham and Bill caught the shuttle bus and went to Heaven's new data processing center. It was a truly huge facility, a hundred times bigger than the Astrodome.

Workmen were crawling all over the place, getting the miles of fiber optic cables properly installed. But the center was dominated by the computers. Half a million computers, arranged neatly row-by-row, half a million ....

   Macintoshes

... all running Claris software! Not a PC in sight! Not a single byte of Microsoft code!

The thought of spending the rest of eternity using products that he had spent his whole life working to destroy was too much for Bill. "What about PCs???" he exclaimed. "What about Windows??? What about Excel??? What about Word???"

'You're forgetting something," said Abraham. "What's that?" asked Bill plaintively.

"This is Heaven," explained Abraham. "We need a computer system that's heavenly to use. If you want to build a data processing center based on PCs running Windows, then....

..GO TO HELL!"

The End of the World is Near!

One day God was looking over creation and He decided that He wasn't really happy with the way things turned out. So He called the 3 most powerful men on earth, Bill Clinton, Boris Yeltsin and Bill Gates, to come and see Him. He told them that this experiment with life on earth was a failure, and that in 3 days He was going to end it.

So basically they had 3 days to prepare their people.

So Boris Yeltsin convenes an emergency meeting of the Russian Parliment and says:

"*I have bad news, and really bad news. First of all, there is a God. Secondly everything we have worked for since the revolution will be totally destroyed in 3 days.*"

Bill Clinton makes a State of the Union address to the American people on TV and says: *"I have good news and bad news. First of all, there is a God. Secondly, everything we have worked for since the revolution will be destroyed in 3 days."*

Bill Gates convenes a meeting of the board of directors and says: "I have good news, and really good news. First of all, there is a God, and He spoke to me personally.

Secondly, in 3 days, IBM will be destroyed."