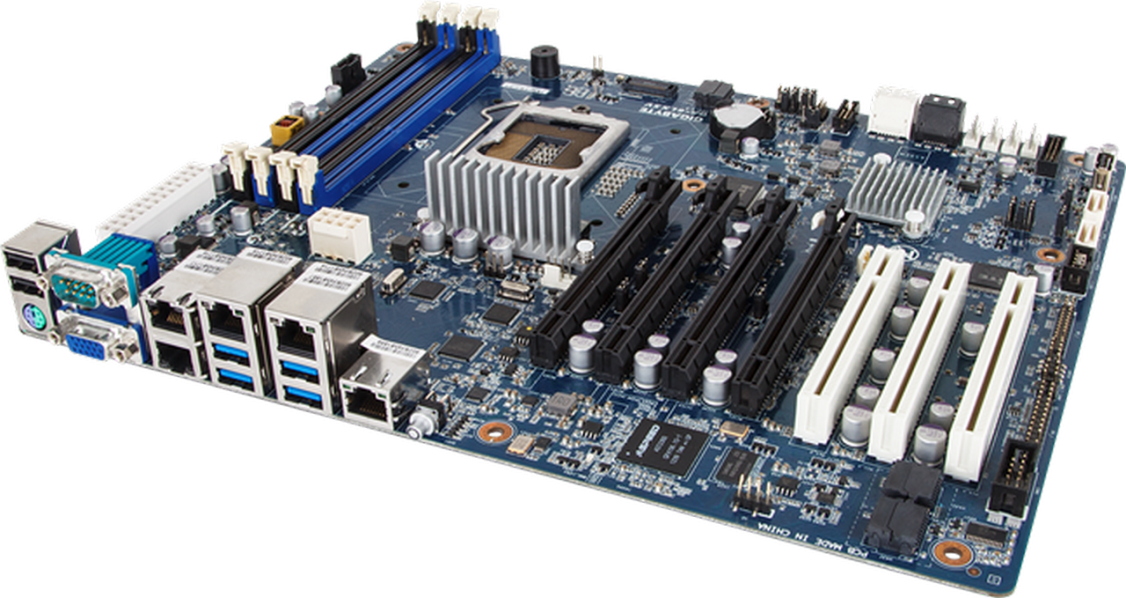


Computing Science

Computer Systems



Computer Structure

Computer Structure means the architecture of the different components that make up a computer system and how they relate to each other.

**Processor**

**Memory**

Address Bus

Data Bus

This can be represented by the following diagram.

We will now look at the above computer system and find out in detail more about the names and function of each part.

Processor

The processor (sometimes known as the central processing unit -CPU) is the part of a computer system that carries out the instructions of a computer program, to perform the basic arithmetical, logical, and input/output operations of the system. It is sometimes known as the brains of the computer.

**ALU**

**Control**

**Unit**

**Registers**

Processor

The processor is divided into three parts : **the control unit**, the **Arithmetic and Logic Unit (ALU)** and **registers**.

* The Arithmetic and Logic Unit performs all the calculations (e.g. + - / \*) and Logical operations (e.g. AND, OR).
* The Control Unit controls the sequencing of fetching, decoding and execution of instructions.
* The Registers are temporary memory locations within the processor itself.

Memory

Memory is a set of chips which store programs and data in the computer.

**1000**

**1001**

**23**

**1002**

**1003**

**1004**

**A**

**1005**

Memory Locations With Unique Addresses

All memory is divided into storage locations.

*Each storage location is given a unique address so that the computer can read and write the data to that location correctly.*

In the diagram opposite, memory location 1001 is storing the number 23. The memory location 1004 is storing A.

Buses

In a computer, all the program instructions are stored in memory and transferred to the processor one instruction at a time to be executed. To do this there must be connections between the processor and memory. These connections are known as buses.

**Processor**

**Memory**

Address Bus

Data Bus

*A bus is a group of wires used collectively to transmit information.*

There are 2 buses that connect the processor and memory.

* *The* ***Address Bus*** *identifies the memory location that is going to be accessed.* It is only a one-way bus as only the processor can identify the memory address.
* *The* ***Data Bus*** *transfers data between the processor and memory and vice versa.* It is a two-way bus as information can be written from the processor to memory and read from memory to the processor.

Languages and Translation

A computer program is a set of instructions which tells the computer what to do.

Here are two programs :

Program 1 Program 2

PRINT "Welcome to my program " 10010010 11011011

PRINT "What is your name " 10011101 01011001

Which program is easier for you to understand? Which program does the computer execute?

High Level Language

Program 1 is an example of a high level language. The common features of all high level languages are:

* It uses English like words
* It has to be translated into machine code
* It can work on different types of computer systems with only minor changes.

Later you will be learning to use the high level language TrueBasic.

Machine Code

Program 2 is an example of machine code. This is the language that the computer executes. The common features of machine code are:

* It is made up of 1s and 0s
* It is machine specific. A machine code program will only run on the type of computer it was created for, not others.

Translation

It should be obvious that if we write a program in a high level language and the computer can only execute machine code then the high level language program must be translated into machine code. This is similar to a French person talking to you in French, you must translate what they are saying before you understand the message.

The computer needs a program which can translate a high level language into its own language. This program is called a translator.



There are 2 different types of translator

• Interpreter

• Compiler

Interpreter

An interpreter translates the high level program one line at a time into machine code and executes this line of machine code immediately. Because you can't save the machine code, this translation must be carried out every time the program is run, therefore interpreted programs are slow.

Two advantages of an interpreter are:

* It reports errors at the end of each line so it is easier to correct your mistakes.
* It takes less memory than a compiler when translating the program into machine code.

One disadvantage of an interpreter is:

* Because the machine code cannot be saved, the high level language program needs to be translated into machine code every time the program is run. This takes time.

Compiler

A compiler translates the whole high level language program (source code) into machine code (object code) **and saves the machine code version of the program**. This machine code program can then be run in future without having to translate it again. This saves time.

Two advantages of a compiler are:

* Because the machine code file is saved, it never needs to be translated again.
* It takes less memory when executing the machine code program than an interpreter.

One disadvantage of a compiler is:

* It takes more memory when translating the program compared to an interpreter.

Memory Use of Interpreters and compilers

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Interpreter | | |  | Compiler | | |
|  | Source Code | Translator | Machine Code |  | Source Code | Translator | Machine Code |
| Translating | ✓ | ✓ |  |  | ✓ | ✓ | ✓ |
| Executing | ✓ | ✓ |  |  |  |  | ✓ |

Programmers use both interpreters and compilers.

* They use interpreters when they are developing the program because each error is reported at the end of each line so debugging mistakes is easier.
* They use compilers once the program is finished so that they do not need to translate it again.

Questions on Pages 2-6

|  |  |  |
| --- | --- | --- |
| 1. | Name and describe the 3 parts of the processor | 6 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| 2. | Describe how the computer can accurately read/write to any memory location. | 2 |
|  |  |  |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| 3. | Complete the diagram below showing the structure of the computer. | 4 |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| 4. | Explain what is meant by a ‘bus’ in Computing. | 1 |
|  |  |  |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| 5. | Describe the purpose of both the address and data buses. | 2 |
|  | Address Bus: |  |
|  |  |  |
|  |  |  |
|  | Data Bus: |  |
|  |  |  |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| 6. | Describe two common features of high level languages. | 2 |
|  |  |  |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| 7. | Explain why a high level language program must be translated. | 1 |
|  |  |  |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| 8. | Describe how an interpreter translates a high level language program. | 1 |
|  |  |  |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| 9. | Describe how a compiler translates a high level language program. | 1 |
|  |  |  |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| 10. | Describe one advantage and one disadvantage of translating a program using an interpreter. | 2 |
|  | Advantage: |  |
|  |  |  |
|  |  |  |
|  | Disadvantage: |  |
|  |  |  |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| 11. | Describe one advantage and one disadvantage of translating a program using a compiler. | 2 |
|  | Advantage: |  |
|  |  |  |
|  |  |  |
|  | Disadvantage: |  |
|  |  |  |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| 12. | Andrew is in the middle of writing his program. He could either use a compiler or interpreter to translate the program. Which would you suggest he uses and give a reason for your answer. | 2 |
|  | Translator: |  |
|  | Reason: |  |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| 13. | Andrew program is now finished and working correctly. Which translator would you suggest he uses and give a reason for your answer. | 1 |
|  | Translator: |  |
|  | Reason: |  |
|  |  |  |

Data Representation

Everything stored by a computer is stored using Binary. In binary there are only two numbers 0 and 1. (This is different from the denary system humans use as it has 10 numbers 0123456789).

Computers use binary to store and process information in memory. Binary is the only thing a computer can execute and for that reason is sometimes known as Machine Code. Anything else that we put into the computer e.g. text, number, picture etc. must be translated into machine code before the computer can understand it.

The smallest unit of memory a computer deals with is called a Bit (Binary DigIT) and can have either of two values 0 or 1.

One bit though only stores a very small piece of information. The computer groups bits together in a series of eight bits called a byte.

1 bit = 0 or 1

8 bits = 1 byte

1024 bytes = 1 kibibyte (Ki)

1024 kibibytes = 1 mebibyte (Mi)

1024 mebibytes = 1 gibibyte (Gi)

1024 gibibytes = 1 tebibyte (Ti)

1024 tebibytes = 1 pebibyte (Pi)

Example 1 - How many bytes are in 3 gibibytes?

3 x 1024 x 1024 x 1024 = 3221225472 bytes

Example 2 - How many mebibytes are in 20971520 bits?

20971520 / 8 / 1024 / 1024 = 2.5 mebibytes

Computers store five types of data in memory: integer numbers, real numbers, characters, bit-mapped and vector graphics.

Because the computer only understands machine code, the binary system is used to represent these different types of data.

Representation of Positive Integers (Convert Binary to Denary)

An integer is a positive or negative whole number. Positive integers are stored in the computer using their place values. This is not something new; you have been using it since primary school. e.g.



**Denary**

The computer of course only uses 0s and 1s. Therefore the place values are different.



Here are three binary numbers for you to convert into denary.

All working must be shown.

1. 

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_=\_\_\_\_\_\_\_\_\_\_

2. 

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_=\_\_\_\_\_\_\_\_\_\_

3. 

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_=\_\_\_\_\_\_\_\_\_\_

Convert Denary to Binary

To convert a denary integer into its binary equivalent you follow these instructions. We will use the example of 85.

1. Draw out your place values table for binary.

128

64

32

16

8

4

2

1

2. Working from the left, work out which is the smallest number that will divide into your denary number once. i.e. 128 will not go into 85 once therefore we put a 0 underneath the 128. 64 will divide into 85 once therefore we put a 1 under the 64. This gives us:

128

64

32

16

8

4

2

1

0

1

3. We have now used 64 of our 85 leaving us 21 (85-64 = 21). We simply repeat step 2. until we have nothing left. i.e. 32 will not divide into 21 once therefore we put a 0 under the 32. 16 will divide into 21 once so we put a 1 under the 16. This gives us:

128

64

32

16

8

4

2

1

0

1

0

1

4. We have now used 16 of our 21 leaving us 5 (21-16 = 5). 8 will not divide into 5 once therefore we put a 0 under the 8. 4 will divide into 5 once so we put a 1 under the 4. We have now used 4 of our 5 leaving us 1 (5-4 = 1). 2 will not divide into 1 once therefore we put a 0 under the 2. 1 will divide into 1 once so we put a 1 under the 1.This gives us:

128

64

32

16

8

4

2

1

0

1

0

1

0

1

0

1

Note that you can check your answer very easily.

128

64

32

16

8

4

2

1

0

1

0

1

0

1

0

1

Simply convert the binary number into denary i.e. 1 + 4 + 16 + 64 = 85.

Here are three denary integers for you to convert into binary.

a) Show how 46 would be stored as a binary number.

128

64

32

16

8

4

2

1

b) Show how 97 would be stored as a binary number.

128

64

32

16

8

4

2

1

c) Show how 29 would be stored as a binary number.

128

64

32

16

8

4

2

1

Representation of Real numbers

A real number is a number that can have a decimal/fractional part to it. To represent a real number in binary we use *floating point representation*. Floating point representation stores the number in the *mantissa* and *exponent*.

To work out the mantissa and exponent you need to :

* move the binary point all the way to the left
* the entire number without the binary number is the mantissa
* the number of places the binary point was moved to the left (expressed as a binary number) is the exponent.

Example 1 - How would 1101.0011011101 would be represented in binary.

Mantissa - 11010011011101 Exponent - 100bin

Note the exponent is 100 as the binary point was moved 4 places and the binary of 4 = 100

Show how 10100.1111 would be represented in binary.

mantissa \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ exponent \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Show how 11.1101 would be represented in binary.

mantissa \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ exponent \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Characters

Characters are stored using its ASCII code. Each character is given a unique number and stored using 8 bits of memory e.g. A = 01000001 in binary, D = 01000100 in binary

ASCII (American Standard Code for Information Interchange) is a standard code for text which allows text to be easily exchanged between different applications and different types of computers.

How many bits and bytes are in the following pieces of text:

|  |  |  |
| --- | --- | --- |
| ***Text*** | ***Bits*** | ***Bytes*** |
| Character |  |  |
| Computer Systems |  |  |
| Now is the time for all good men to come to the aid of the party. |  |  |

Use the computer to find the ASCII code for:

|  |  |  |
| --- | --- | --- |
| ***Character*** | ***Denary*** | ***Binary*** |
| A |  |  |
| a |  |  |
| 3 |  |  |
| é |  |  |
| tab |  |  |

The extended ASCII table includes:

* all the numbers, letters (upper and lower) and punctuation marks in the alphabet
* foreign language symbols e.g. é, ö, ê
* control characters e.g. tab, line feed, carriage return

Bit-mapped Graphics

A bit-mapped graphic is stored as a 2-dimensional array of pixels where each pixel stores the colour of that pixel. Normally you don't see the pixels as you only look at the overall picture. Here is a rabbit coming out of a hat. You don't see many dots, but when it is magnified, all the pixels can be seen clearly.

Black and white computer graphics are represented in binary by: if the pixel is black then a 1 is stored in a bit, if it is white then a 0 is stored.



Each pixel in a black and white bit-mapped graphic is stored in 1-bit of memory. By counting the pixels, you can work out how much memory it takes to store the picture. The above graphic takes 88 bits of memory.

Colour graphics require more than 1-bit of memory to store a pixel.

* 8-bit graphics require 8 bits of memory to store the colour of each pixel.
* 24-bit graphics require 24 bits of memory to store the colour of each pixel.

00

01

01

00

01

10

10

01

01

10

10

01

00

01

01

00

00 = white

01 = black

10 = grey

This would significantly increase the amount of memory required to store a graphic.

The example opposite uses 2-bit colour i.e. it takes 2-bits per pixel to store the colour of the pixel.

Vector Graphics

Vector graphics store a picture by storing each object’s attributes. A circle might be stored as:

**circle(centrex, centrey, radius, line colour, line thickness, fill colour)**

object

attributes

i.e. it doesn’t store the picture itself but the instructions of how to draw the picture.

A line might be stored as

**line(startx, starty, end x, endy, line colour, fill colour).**

Obviously, when stored in the computer each attribute would be stored as a binary number.

There are several common objects that you need to know how they are represented. These are:

* Rectangle
* Ellipse
* Line
* polygon.

For each of these you need to know how the attributes of:

* co-ordinates
* line colour
* fill colour.

Rectangle:

Rectangle(height, width, x, y, fill colour, line colour)

For example:

rectangle(2, 5, 2, 3, grey, black) would draw:

|  |  |  |
| --- | --- | --- |
| Height | the height of the rectangle |  |
| Width | the width of the rectangle |
| X,Y | the starting coordinates of the rectangle |
| Fill colour | the colour the rectangle will be filled with |
| Line colour | the colour of the line the rectangle is drawn with |

Line:

line(x1, y1, x2, y2, line colour)

For example: line(1, 2, 6, 7, black) would draw:

|  |  |  |
| --- | --- | --- |
| x1, y1 | Coordinates of beginning of line |  |
| x2, y2 | Coordinates of end of line |
| line colour | the colour of the line |

Ellipse:

ellipse(cx, cy, rx, ry, fill colour, line colour)

For example:

ellipse(5, 5, 4, 2, grey, black) would draw:

|  |  |  |
| --- | --- | --- |
| cx, cy | Centre point of the ellipse |  |
| rx | Horizontal radius of ellipse |
| ry | Vertical radius of ellipse |
| Fill colour | the colour the ellipse will be filled with |
| Line colour | the colour of the line the ellipse is drawn with |

Polygon:

polygon(x1, y1, x2, y2, x3, y3, fill colour, line colour)

For example: polygon(1, 3, 3, 7, 5, 3, grey, black) would draw:

|  |  |  |
| --- | --- | --- |
| x1, y1 | Coordinates of first point of polygon |  |
| x2, y2 | Coordinates of second point of polygon |
| x?, y? | Coordinates of next point of poygon. E.g. a pentagon would have 5 points |
| Fill colour | the colour the ellipse will be filled with |
| Line colour | the colour of the line the polygon is drawn with |

Questions on Pages 10 - 19

1. Carry out the following calculations.

a) Put the following memory units in ascending order. gibibyte, kibibyte, pebibyte, bit, byte, tebibyte.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

b) Show how the binary number 1001 would be stored in the computer.

c) Show how the denary number 37 would be stored in a computer.

d) Show how the binary number 101.10111 would be stored in a computer.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

e) Explain how bit-mapped graphics are represented in memory.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

f) Explain how vector graphics are represented in memory.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

g) Describe how characters are stored in computer memory.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Below is shown a simple black and white graphic.



h) In the picture above, complete the bit-map for the graphic.

i) How many bits would it take to store the above graphic?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

j) How many bytes would it take to store the above graphic? Show working.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

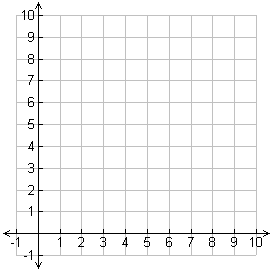
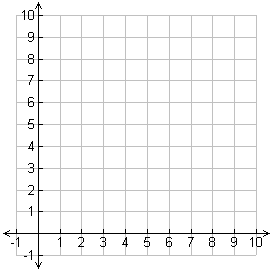
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

k) Explain what would happen to the file size if the above graphic had been drawn using 8-bit colour.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

l) Write the vector graphics code that will draw **both** the diagrams shown below.



\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

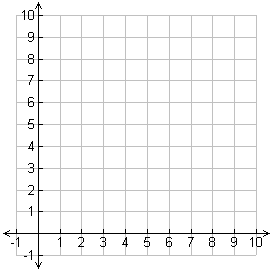
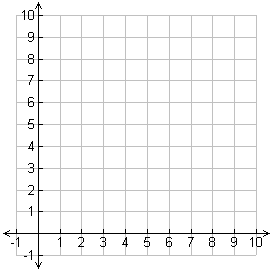
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

m) Draw the graphics that would be drawn using the following code.

polygon(3, 3, 3, 8, 7, 8, 7, 3, grey, black)

rectangle(5, 4, 3, 3, grey, black)



Environmental Impact

When we are using a computer, we are using electricity, not only for the computer, but also the monitor, printer, wireless router etc. This energy use can have a detrimental impact on the environment.

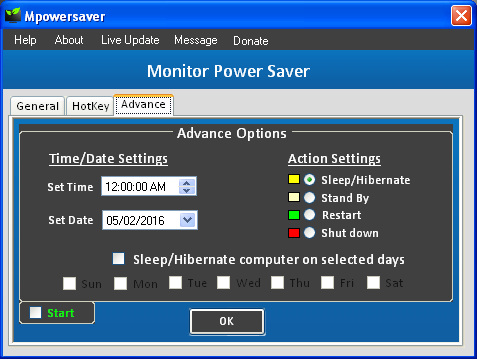
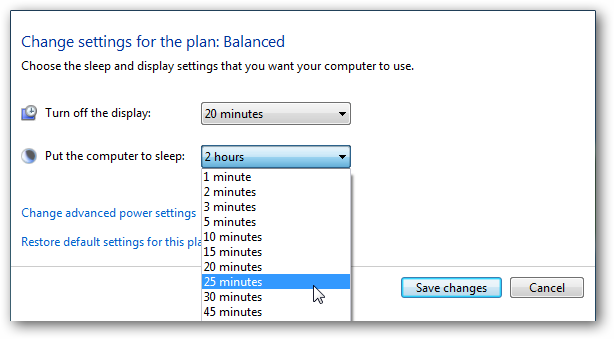
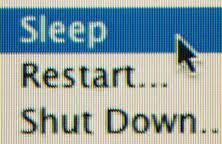
However, computers reduce environmental impacts. For example, sending an e-mail to USA.

Obviously while using the computer to send the e-mail, you are using up energy and when your computer was being made a lot of energy would have been used and a lot of carbon dioxide produced.

However, consider the alternative: trees would be cut down to produce the paper to write on. The letter would be picked up by a post office van burning fuel. It would be taken to the sorting office, then driven by van to the airport, more fuel. The plane would then burn lots of fuel taking it to USA where once again it would be loaded onto vans, driven to their sorting offices, then driven to homes using an awful lot of fuel.

Reducing Computer’s Environmental Impact

Some of the ways we can reduce the environmental impact is by:

1. ***Leaving computers on stand-by***  
   If you are going to be using the computer intermittently, then set the stand-by mode to come on after 20 minutes of inactivity. This will reduce the power consumption quite considerably.
2. ***Choose the settings on your monitor carefully***  
   The monitor settings can also save quite a lot of energy. To reduce the energy impact:
   1. Set the monitor to sleep/stand-by after 20 minutes of inactivity will save a lot of energy.
   2. Setting the brightness of the monitor to a dimmer setting will save power. (Having the monitor set to very bright will use more power than having it set to a dimmer setting).
3. ***Power Down Settings***You are able to choose the settings that will power down the computer.
   1. Changing the settings to a shorter time of inactivity should save energy (but may become a nuisance if you set it for too short a time).
   2. It was suggested that 50% of the power used by a computer is when the user is not using it! For example, leaving it on all night. You can set a time for the computer to shut down and switch on. (The school computers switch off at 18.30 and switch back on at 07.00).

Using all three of these methods should reduce your power usage and hence the environmental impact of using your computers.

Security Precautions

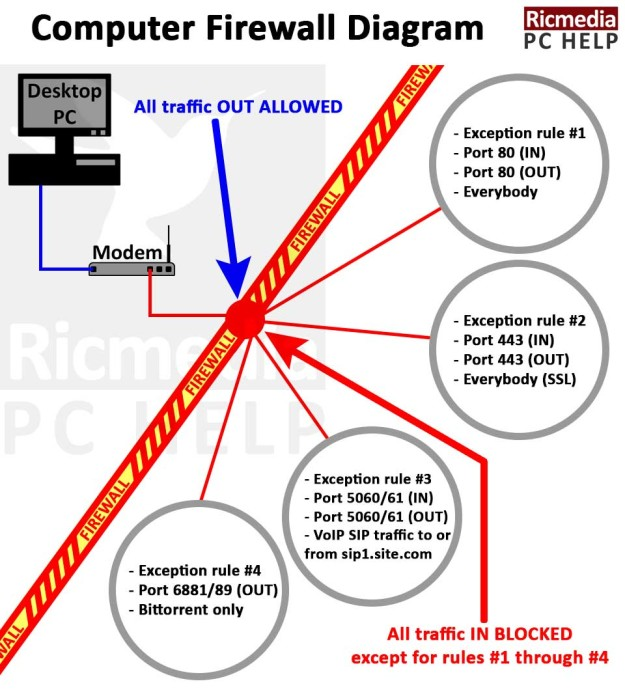
There are a large number of threats to your computer and the information stored on it. For example: viruses, hacking, phishing etc.

To try and protect against threats like these there are some basic precautions you should take:

Firewall

A firewall protects against people trying to hack into your computer. It does this by:

When a external computer tries to access your computer, the firewall examines whether it is allowed to access your computer. If it is, then access is given. If not then it is blocked.  
  
Ways in which a firewall can block access

* ***Packet filtering***When a data packet is received by the firewall, it checks it against its filters. If it is acceptable it is allowed through otherwise it is rejected.
* ***IP Blocking***  
  *When a specific IP address tries to communicate with your computer, the firewall can compare the IP address that it came from against a list of blocked IP addresses. If it is in the blocked list then it is rejected (alternatively it can be checked against a list of IP addresses that are allowed to access the computer).*

Encryption

More and more information is being transferred electronically. Much of this information is private and confidential and if found out by criminals could be used against us. For example, when shopping on the Internet, if someone intercepted your credit card details then they would be able to buy goods using your credit card details.

**Source**

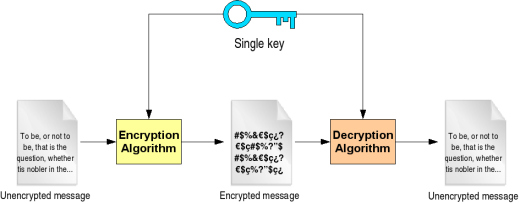
**Destination**

**Intercepted**

One solution to this problem is to encrypt the information before it is sent electronically.

*Encryption codes the information so that if it is intercepted then they would not be able to make sense of it.*

At the senders side the message is encrypted using a ‘key’ and at the destination end it is decrypted using a similar key.



Questions on Pages 23 – 26

|  |  |  |
| --- | --- | --- |
| 1. | Describe three ways how the energy use of computer systems could be reduced. | 3 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| 2. | Describe the role of the firewall. | 1 |
|  |  |  |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| 3. | Describe two ways in which a firewall operates. | 2 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| 4. | Explain what is meant by ‘encryption’ in Computing. | 1 |
|  |  |  |
|  |  |  |
|  |  |  |
| 5. | Describe the process of sending and receiving an encrypted message. | 2 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Acknowledgements

http://www.cdm.depaul.edu/academics/PublishingImages/heroMSCS.jpg

http://www.pngall.com/wp-content/uploads/2016/04/Motherboard-PNG-Picture.png

https://www.howtogeek.com/wp-content/uploads/2014/08/xcpu-on-motherboard.jpg.pagespeed.gp+jp+jw+pj+ws+js+rj+rp+rw+ri+cp+md.ic.tL1N0E5yoS.jpg

http://ww1.prweb.com/prfiles/2009/07/21/1584114/gI\_0\_PSSmokestack.jpg

http://mpowersaver.in/images/Print3.PNG

http://pchelp.ricmedia.com/wp-content/uploads/computer-network-firewall/computer-network-firewall-diagram-625x690.jpg

http://www.ccfit.nsu.ru/~ask/grid/progtutorial/images/security\_concepts\_symmetric.png