IB DIPLOMA PROGRAMME
PROGRAMME DU DIPLÔME DU BI
PROGRAMA DEL DIPLOMA DEL BI

Computer science
Higher level and standard level

Specimen paper 1s and paper 2s

For first examinations in 2006

CONTENTS

IB DIPLOMA PROGRAMME
PROGRAMME DU DIPLÔME DU BI
PROGRAMA DEL DIPLOMA DEL BI

COMPUTER SCIENCE
HIGHER LEVEL
PAPER 1

SPECIMEN

2 hours 15 minutes

INSTRUCTIONS TO CANDIDATES

- Do not open this examination paper until instructed to do so.
- Section A: answer all the questions.
- Section B: answer all the questions.

SECTION A

*Answer **all** the questions.*

1. Outline one example of a *real-time processing* system. *[2 marks]*

2. (a) Explain the function of *cache memory*, as it relates to the functions of the *CPU*, the *RAM*, and the *data bus*. *[3 marks]*

(b) Outline the function of an *address bus*. *[2 marks]*

3. (a) State **two** types of *language translators*. *[2 marks]*

(b) Outline the function of high level language translators. *[2 marks]*

4. Define the term *prototyping*. *[2 marks]*

5. Draw the representation of the *binary search tree* if the following data were inserted in this order:

FALCON, CANARY, PIGEON, TURKEY, OSPREY. *[3 marks]*

6. Draw the *logic circuit* that corresponds to the following *Boolean expression*.

$$A + \overline{B} \cdot C$$ *[3 marks]*

7. An organization has to determine how much security their computer system requires. State **two** factors that should be considered in this determination. *[2 marks]*

8. (a) Outline **two** essential *hardware* components of a Local Area Network. *[2 marks]*

(b) With the aid of a diagram or otherwise, describe a *star* network. *[3 marks]*

9. Outline the need for a *data compressor*. *[2 marks]*

10. (a) Describe a suitable method to represent signed integers using **two** bytes. *[2 marks]*

(b) Illustrate your answer by stating in hexadecimal notation how the integer numbers
(i) 32 *[1 mark]*
(ii) -32 *[1 mark]*
would be stored.

11. Outline the basic features of:

(a) *Polymorphism* *[2 marks]*

(b) *Inheritance*. *[2 marks]*

12. By tracing the following recursive algorithm

```
public int calc(int n)
 {
   if (n<2)
   { return n; }
   else
   { return (n -1)*calc(n-1); }
  }
```

calculate `CALC(4)`. All steps and calls must be shown clearly. *[4 marks]*

SECTION B

*Answer **all** questions.*

13. At an Office Supplies Store, customer order data can be received as follows:

- A customer may send an order in the mail
- A customer may stop by the store, pick out the needed items from the shelves and bring them to the POS terminal.

(a) Describe **one** possible way data can be collected for **each** circumstance. *[4 marks]*

Order data received is stored in a transaction file held on disk. Every day the transaction file is sorted and the sorted transaction file is used to update the inventory master file held on disk. New inventory master file is created and different reports printed.

(b) Construct a system flowchart that illustrates the update of the inventory file. *[5 marks]*

(c) State **one** type of report that can be generated. *[1 mark]*

14. (a) Outline **three** functions of *operating systems*. *[3 marks]*

(b) (i) Explain how a user of a standalone microcomputer could use the facility provided by *multitasking*. *[2 marks]*

(ii) State **one** disadvantage in using a *multitasking* operating system on a single user computer. *[1 mark]*

An operating system is written which provides a library of *classes* for software developers to use. All software packages designed to run on the operating system have a consistent human computer interface.

(c) Outline **two** advantages to software developers of accessing methods from a *library class*. *[2 marks]*

(d) State **one** advantage to users and **one** advantage to developers of using a consistent human computer interface. *[2 marks]*

15. Given the following algorithm fragment

Algorithm 1:

```
int n=4;
for(int k = 0; k<n; k=k+1)
{
   ndata[k] = 0;
   for (int j =0; j<k+1; j=j+1)
   {
      ndata[k] = ndata[k] + data[j];
   }
}
```

and the array `data`

[0]	[1]	[2]	[3]
1	3	2	5

(a) By tracing the algorithm or otherwise, outline the contents of array `ndata` after execution of the algorithm. *[3 marks]*

Algorithm 2 solves the same task as Algorithm 1

Algorithm 2:

```
int n = 4;
ndata[0] = data[0];
for(int k = 1; k<n; k=k+1)
{
     ndata[k] = ndata[k-1] + data[k];
 }
```

(b) (i) Discuss the efficiency of Algorithm 1 and Algorithm 2, and state the *BigO* notation of both algorithms. *[5 marks]*

(ii) Outline **two** reasons why analysing an algorithm by studying the running times of a specific implementation should be avoided. *[2 marks]*

16. (a) Define the term *interrupt*. *[2 marks]*

When transferring data to an *input/output device*, the computer system uses *interrupts*.

(b) Outline **two** other examples of *interrupts* in a computer system. *[2 marks]*

(c) Explain the purpose of a *buffer* during input/output transfers. *[2 marks]*

(d) Explain *double buffering*. *[2 marks]*

(e) Explain how a transmission error can be detected by *parity checking*. *[2 marks]*

17. Given the following *dynamic queue*.

(a) Construct a diagram showing the representation of how the queue would look after adding FALCON and CANARY (in this order). *[3 marks]*

(b) State the order in which the data will be deleted from the queue. *[2 marks]*

(c) Construct a diagram showing a representation of a *dynamic stack* after execution of the following algorithm fragment.

```
stack.push('PIGEON');
stack.push('CANARY');
first = stack.pop();
second = stack.pop();
stack.push ('FALCON');
stack.push('TURKEY');
```

[3 marks]

(d) State the value of the popped variable

(i) `first` *[1 mark]*

(ii) `second`. *[1 mark]*

18. A college occupies a building on three levels. A computer network is in place but needs extending. It is suggested that the current system be replaced by a wireless network which gives connection to a file server and a separate gateway out to the Internet.

(a) Outline the hardware features which would have to be considered in a *feasibility study* for the new system. *[2 marks]*

(b) Discuss **one** advantage and **one** disadvantage of allowing students to link up to the network with their personal laptops. *[4 marks]*

The college receives a large file from a centre of education in another country. The sending machine has a different operating system from that of the college.

(c) Explain the role of *packets* in the sending and receiving of this file. *[4 marks]*

MARKSCHEME

SPECIMEN

COMPUTER SCIENCE

Higher Level

Paper 1

SECTION A

1. *Award **[2 marks]** for any of the following, **[1 mark]** for a vague answer.*
air traffic control system (that tracks the flight path of airplanes near an airport);
monitoring (of patients in hospital intensive care); ***[2 marks]***

2. (a) *Award **[3 marks]** for any explanation which includes a correct relationship to each of the other components, e.g. **[1 mark]** for CPU, **[1 mark]** for RAM, **[1 mark]** for bus, for example.*
the data bus transfers data from the RAM to the cache;
the CPU retrieves data directly from the cache, without accessing the RAM;
RAM contains a lot of data, but cache only retrieves a small amount at a time; ***[3 marks]***

*It is not necessary to describe technical details such as data-transfer-rates, paging, write-through, etc. A diagram showing the cache sitting between the RAM and CPU should be awarded **[2 marks]** if everything is labelled but no further explanation is provided.*

*Award **[3 marks]** for any explanation which includes a correct relationship.*

(b) *Award **[1 mark]** for each of the following.*
the address bus carries the address of data or instructions;
to be transferred between memory and CPU;
giving access to the location required; ***[2 marks max]***

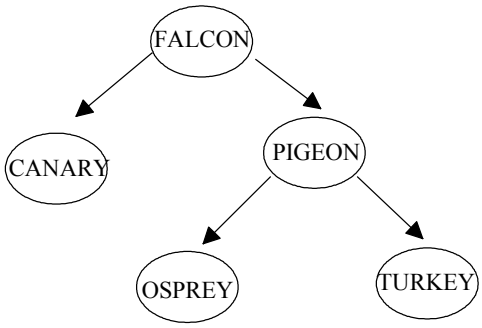
3. (a) *Award **[1 mark]** for each of the following.*
interpreter;
compiler;
assembler; ***[2 marks max]***

(b) to convert high level instructions into machine language; ***[2 marks]***

4. *Award **[1 mark]** for*:
the construction of a simple version of a system in the design stage;

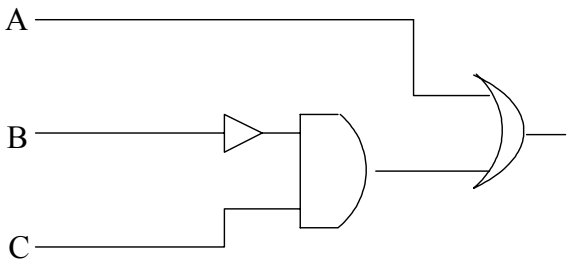
*Award **[1 mark]** for either of the following points (or something resembling)*:
showing the user interface but without full processing behind it;
allowing the user to propose changes at the design stage; ***[2 marks max]***

5. *Award **[1 mark]** for correct root, **[1 mark]** for each correct subtree.*
Accept the mirror image.



[3 marks]

6. *Award **[1 mark]** for each gate – not, and, or – at correct position.*



[3 marks max]

7. *Award **[1 mark]** for any appropriate factor. For example*
the value of hardware;
the value of software;
the cost of controls;
the cost of replacing hardware;
the cost of new software, *etc.*; ***[2 marks max]***

8. (a) *Award **[1 mark]** for each of the following.*
workstations, terminals;
servers (print server, file server);
components that can be shared by users;
connection hardware (cables, hardware that connects nodes in the network);
network interface cards; ***[2 marks max]***

(b) *Award **[1 mark]** for each of the following.*
involves centralized host computer;
connected to a number of smaller computer systems;
which communicate with one another through host;
and usually share the host's computer data base; ***[3 marks max]***

9. *Award **[1 mark]** for each of the following.*
to shorten the length of files;
by elimination all unnecessary data – empty records, redundancies;
to speed up uploads;
to put on a floppy disk *etc*.; ***[2 marks max]***

10. (a) Two's complement

sign bit ↑ 2 bytes = 16 bits, integer range: from -2^{16-1} to $2^{16-1}-1$

[2 marks]

Award marks for other representations – one's complement or sign & magnitude).

(b) *Answers are given for two's complement representation. Award marks for the correct representation using the method described in part (a).*

(i) 0020; ***[1 mark]***

(ii) FFE0; ***[1 mark]***

11. (a) *Award **[1 mark]** for each of the following.*
same operation of identifier can be applied;
to different objects or data types; ***[2 marks]***

(b) *Award **[1 mark]** for each of the following.*
an object extends an existing object;
taking the data members and methods from the original; ***[2 marks]***

12.
```
CALC(4)
      =3*CALC(3);
      =3*2*CALC(2);
      =3*2*1*CALC(1);
      =3*2*1*1=6 ;
```
[4 marks]

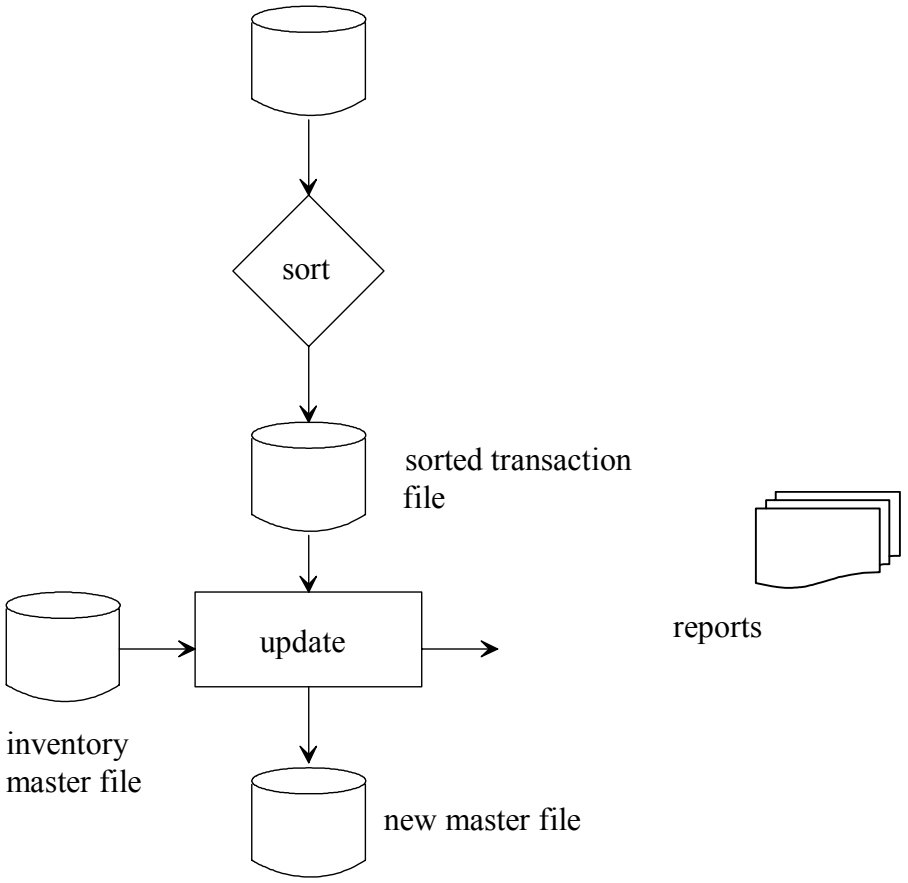
SECTION B

13. (a) *Award **[1 mark]** for identifying a method, **[3 marks]** for further explanation.*

records received by mail would be collected, grouped into small batches; and entered all at once into the system;

orders placed at the store can be handled in several ways, for example: if all the items have the bar coded tags, bar code reading devices can be used to automatically scan the data into computer usable form; **OR** simply the keyboard at the counter can be used to enter the data bar code on the items ordered if not possible to read bar code; ***[4 marks]***

(b) *Award **[1 mark]** sort trans file, **[2 marks]** for update moving across two correct inputs, **[1 mark]** for update producing output and **[1 mark]** for update producing master file.*



[5 marks]

(c) *Award **[1 mark]** for any of the following.*
customer invoices;
summary of products sold;
summary of products ordered;
periodical status report (of all goods in stock);
summary of invoice related data – discounts, unpaid customer invoices, *etc.*;
reorder reports – list of items with the quantity below the minimum stocking level *etc.*; ***[1 mark max]***

14. (a) *Award **[1 mark]** for identification and some elaboration for each of the following. If no elaboration is given but three valid points identified then award **[2 marks]**.*
coordinating jobs;
memory management;
backing storage management;
scheduling;
resource allocation;
I/O control;
interrupt handling;
security;
accounting *etc.*; ***[3 marks max]***

(b) (i) *Award **[2 marks]** for any description that includes one user able to perform/switch between several tasks at the same time.*
typing test using word processing program (first window), create picture (second (window), and combine the two in DTP package (third window);
writing a user guide to a program, one window can be used for writing guidelines the other can be used to run the program *etc.*; ***[2 marks]***

(ii) *Award **[1 mark]** for any of the following.*
more memory needed;
computer run slower;
if one task running crashes, the other one will also crash; ***[1 mark max]***

(c) *Award **[1 mark]** for each of the following.*
development time will be shortened (because pre-written and pre-compiled routines can be used;
able to make good use of good ideas;
routines are already tested *etc.*; ***[2 marks max]***

(d) *For users*:

new software is much easier to learn;
easy switch from one to another package;
able to guess effect of selected command *etc.*;

For developers:

interface classes already exist so no need to rewrite them;
modification to new packages is a simple matter of creating new methods;
testing time reduced; ***[2 marks max]***

15. (a) *Award **[1 mark]** for each bold number.*

NEW

[1]	[2]	[3]	[4]
1	**4**	**6**	**11**

[3 marks]

(b) (i)

ALGORITHM 1

outer loop executes N times, inner loop executes K times for values range from 2 to K
there are

$1+2+...+(N-1)=\frac{N(N-1)}{2}$ additions;

initializes N variables in array `NEW`
and changes their values

$(N-1)=\frac{N(N-1)}{2}$ times

together there are

$\frac{N(N-1)}{2}+\frac{N(N-1)}{2}+N$ operations;

therefore it is $O(N^2)$;

ALGORITHM 2

loop executes N-1 times, there are N-1 additions

it initializes variable `NEW[1]` once

and changes the value if `NEW[K]`

$K=2,3...,N$;

together there are

$(N-1)+1+(N-1)$ operations;

therefore it is $O(N)$;

[5 marks max]

(ii) *Award **[1 mark]** for each of the following.*

running times are influenced by factors such as programming style;
particular computer;
the data on which the program is run (for example unsorted disk, sorted disk *etc.*;

[2 marks max]

16. (a) an *interrupt* is a hardware generated signal which causes suspension of the execution of the program which currently has control of the CPU; ***[2 marks]***

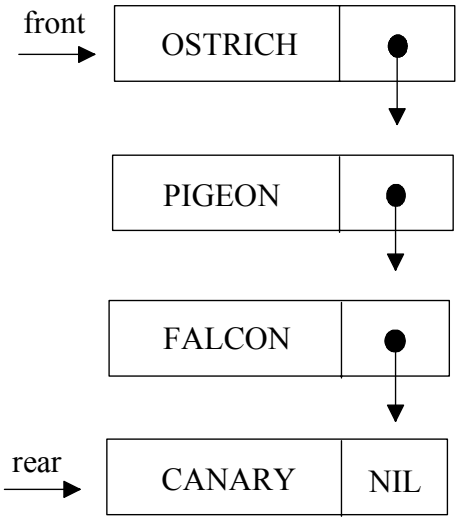
(b) *Award* ***[1 mark]*** *for each of the following.*
program check interrupts;
overflow;
underflow;
division by 0;
external interrupts;
expiry of time slice;
interrupt key pressed by operator;
signal from another processor;
restart;
interrupts caused by malfunctioning hardware *etc*.; ***[2 marks max]***

(c) a buffer is an area of memory used for holding data during I/O transfers;
data to be inputted/outputted is placed in a buffer, the CPU issues the command to start I/O operation and data is transferred to/from the I/O device independently of the CPU; ***[2 marks]***

(d) *Award* ***[1 mark]*** *for each of the following*.
double buffering involves setting aside two areas of memory – to speed up data transfer;
whilst one area is being the emptied – the second is filled; ***[2 marks]***

(e) *Award* ***[1 mark]*** *for each of the following*.
computer can be designed to use an even parity scheme which means that it expects the total number of 1s (in binary code) to always add up to an even number;
a transmission error is signalled by the computer when the number of 1s adds up to an odd number;
The opposite is true when an odd parity scheme is used; ***[2 marks max]***

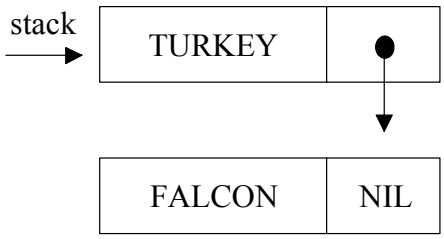
17. (a) *Award **[1 mark]** for CANARY and FALCON at correct position, **[1 mark]** for showing all links and NIL and **[1 mark]** for calculating front and rear pointers.*



[3 marks]

(b) PIGEON, FALCON, CANARY ***[2 marks]***
Look for follow through marks, if the answer matches the drawing in part (a).

(c) *Award **[1 mark]** for stack pointer, NIL and link, **[1 mark]** for correct position of TURKEY and **[1 mark]** for correct position of FALCON.*



[3 marks]

(d) (i) CANARY; ***[1 mark]***

(ii) PIGEON; ***[1 mark]***

18. (a) *For an outline candidates are expected to identify a hardware aspect and briefly mention the details. Award **[1 mark]** for each valid outline e.g.*
the number of hubs needed – probably one on each floor as wireless does not reach that far;
the size of server needed – adequate storage space for the size of the network – adequate processing to cope with multi access;
switch to take directly from hubs to internet;
number of workstations that will be connected at one time is an issue for traffic on the intranet; ***[2 marks max]***

(b) *Award **[2 marks]** for each discussion. One discussion for advantage and one for disadvantage. In each case award **[1 mark]** for a correct point raised and **[1 mark]** for elaborating the consequences e.g.*

Advantages:
students will be able to transfer work directly between college and home without the need for floppy disks *etc*. and without having to send via email. This is more efficient and ensures that they have access to up to date college information;
information to which they have rights of access are not necessarily available on the internet and they can get this in college with ease;
can synchronize their laptops with their files on the server and hence always have the appropriate copy of documents without risking overwriting of the correct one by mistake;

Disadvantages:
for the college, if there are not appropriate virus checking procedures and firewalls in place students could introduce viruses which would be difficult to track down;
may be all sorts of illegal activities such as downloading peer to peer files directly from internet to laptops which should not be allowed by college;
[4 marks max]

(c) *Award **[1 mark]** for each legitimate activity/feature of packets*.
the file will be split up into packets;
each of which will have protocols included;
giving information on destination – and sending addresses;
some data transmission error checking (such as parity *etc*.);
protocol information to allow reception by another operating system;
packets sent separately but labelled in order;
may not arrive in the same order as sent as they will generally take the quickest path which may change during transmission;
reassembled at the other end; ***[4 marks max]***

IB DIPLOMA PROGRAMME
PROGRAMME DU DIPLÔME DU BI
PROGRAMA DEL DIPLOMA DEL BI

SPEC/5/COMSC/HP2/ENG/TZ0/XX

COMPUTER SCIENCE
HIGHER LEVEL
PAPER 2

SPECIMEN

2 hours 15 minutes

INSTRUCTIONS TO CANDIDATES

- Do not open this examination paper until instructed to do so.
- Answer all questions

1. An array, `numbers[]`, has been declared and contains the following values.

49	11	16	72	2	37	14	26

Consider the following code fragment. All variables have been declared.

```
i = 0;
j = 7;
while (i < j)
{    l = i;
     h = j;
     for (c = i; c < j+1; c++)
       {   if (numbers[c] < numbers[l])
               {l = c;}
           else if (numbers[c] > numbers[h])
               {h = c;}
        }
       temp = numbers[i];
       numbers[i] = numbers[l];
       numbers[l] = temp;
       if (l + 1 < j)
        {   temp = numbers[j];
            numbers[j] = numbers[h];
            numbers[h] = temp;
         }
       i++;
       j--;
}
```

(a) By using a trace table, or otherwise, determine the values of h and l after the `c` loop has been completed (for all values of `c`) for the first time. *[2 marks]*

(b) Deduce the purpose of this loop. *[1 mark]*

(c) Copy and complete the following trace table for the whole algorithm.

i	j	l	h	numbers[]
…	…	…	…	..

[4 marks]

(d) State the purpose of the algorithm and outline **two** other methods of obtaining the same result. *[5 marks]*

(e) Construct the algorithm for one of the methods outlined in (d). *[8 marks]*

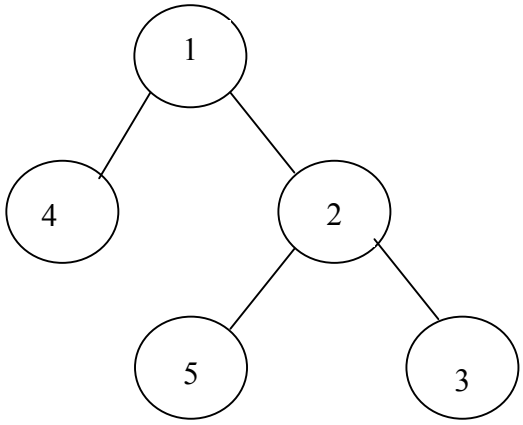
2. A database is devised to hold information about those working for and attending a college. The name and phone number are held for each person and the following class is defined.

```
class Person {

   private String name;
   private String phone;

   public String getName(){
      return name;
   }

   public String getData (){
      return name + phone;
   }

}
```

(a) Using *inheritance*, define a student class which has name, phone and grade stored, and incorporate a method to print out these three details if the student has a grade greater than 3. *[4 marks]*

One feature of the database is that students can be accessed in order of name, by using a binary tree structure. For example,

Record Index	Name
1	Jones
2	Robertson
3	Zhang
4	Collins
5	Lee



(b) Outline a dynamic data structure suitable to hold this data. *[3 marks]*

(c) Using the data structure you defined in (b), construct the recursive method which returns the number of nodes in a binary tree. Make clear any variables that are declared outside the method. *[7 marks]*

(d) Compare the advantages and disadvantages of implementing the tree as a dynamic data structure or as an array. *[6 marks]*

3. Personnel records are stored in a computer file. They are saved and retrieved using a hash table which stores a 4-letter uppercase code in the key field – each employee has a unique code which starts with the department code (*e.g.* A for accounting, M for marketing). The following table converts letters to a numeric value:

A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	10	11	12

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
13	14	15	16	17	18	19	20	21	22	23	24	25

Thus, a record with a key field containing the code ACED would be converted to a hash key as follows:

- look up the numeric values of the letters
- add them together to get the sum
- multiply the sum by 11
- calculate the result of sum **modulo** 50

(a) Determine where the record for ACED would be stored. *[3 marks]*

(b) Determine the maximum number of records that can be stored using this system. *[1 mark]*

(c) Outline **two** problems that could occur using this system. *[4 marks]*

(d) For each of the problems you identified in (c) identify **one** possible solution. *[4 marks]*

(e) It is later required to access this file using another unique field; employee number. Explain how this could be done. *[3 marks]*

An upgrade to the system requires that:

- access to the records is achieved without the use of the look-up table.
- more memory locations are needed.
- input validation is required.
- allow lower case input.

(f) Describe the changes you would make to the program to achieve this. *[5 marks]*

4. *This question requires the use of the Case Study*.

A phrase of music to simulate being played by an orchestra can be reduced to a set of instructions.

(a) Identify **two** items of data that would be processed by the instructions. *[2 marks]*

(b) Outline how these **two** items would be used in the execution of an instruction. *[4 marks]*

(c) Explain the use of *batch processing* and *real time processing* in the use of MIDI devices. *[4 marks]*

(d) Suggest the most suitable file structure for storing the data and explain your choice. *[5 marks]*

(e) With reference to the system flowcharts (figure 6 on page 10 and figure 8 on page 11).

(i) Outline the *input – processing – output* that takes place when more than one MIDI instrument is used to produce a complex musical composition. *[6 marks]*

(ii) Identify **two** features of a home computer that make it suitable to produce professional quality music from input via the keyboard *[2 marks]*

(f) Describe the role of *drivers* in the control of hardware devices, using MIDI as an example. *[4 marks]*

A software writer has been approached to create a program that film makers can use to add music to films. The person employed to do this is called a mixer. The musician has composed suitable music for the film and the mixer is required to add appropriate parts at the right time in the film. The mixer is extremely good at choosing the right passages of music to add to a film and knows when they need to be faded in and out, but she is not a computer expert.

(g) Discuss the advantages of using prototyping during the development stage of the mixer. *[5 marks]*

(h) Suggest a suitable user interface for the mixer and explain how this can be tested out by the program writer. *[4 marks]*

(i) Discuss the implications to professional musicians of the use of MIDI in the creation and playing of music. *[4 marks]*

MARKSCHEME

SPECIMEN

COMPUTER SCIENCE

Higher Level

Paper 2

11 pages

1. (a) A trace table is not asked for but would allow any follow through if need be. If the wrong answer is given but there is a table then award 1 mark for correct loop with no more than two changes to l and h and two marks if there is one error in the changes for l or h.

correct answer: l = 4, h = 3; ***[2 marks]***

(b) the purpose of the loop is to find the highest and lowest values in the array and store them in h and l; ***[1 mark]***

(c)

i	j	l	h	numbers[]
0	7	4	3	(2, 11, 16, 26, 49, 37, 14, 72)
1	6	1	4	(2, 11, 16, 26, 14, 37, 49, 72)
2	5	4	5	(2, 11, 14, 26, 16, 37, 49, 72)
3	4	4	3	(2, 11, 14, 16, 26, 37, 49, 72)
4	3			

*Award **[1 mark]** for each of the following*:
columns i and j;
columns l and h;
numbers column;
termination state of i and j; ***[4 marks]***

(d) *Award **[1 mark]** for*:
algorithm sorts the data into ascending order;

*Award **[1 mark]** for each alternative type of sort and **[1 mark]** for an outline*.
For example:
s*election sort*;
sorts data by finding the smallest and swaps it into the 1st position, finds the next smallest and swaps into 2nd position etc until sorted;
b*ubble sort*;
compares and swaps adjacent items for each pair in the list and after this is completed once the smallest is at the top (or largest at the bottom) this is repeated until all in order;

These are the two on the syllabus and hence the most likely but accept any other outlined (Quicksort, Shellsort etc.) ***[5 marks max]***

(e) For example:

```
public void bubbleSortB(int[] numbers)
{

   int current, temp;
   boolean done;

   do
   {
      done = true ;
      for(current = 0; current < 7; current = current + 1)
      {  if (numbers[current] > numbers[current + 1])
         {
            temp = numbers[current];
            numbers[current] = numbers[current+1];
            numbers[current+1] = temp;
            done = false;
         }
      }
   } while (!done);
}
```

*Award **[1 mark]** for each of the following*:
correct method definition;
array of numbers passed to method;
definition of any variables used;
correct outer loop (can be a for loop);
correct inner loop;
correct test;
correct swap;
early exit (or reduced inner loop);
no syntax mistakes; ***[8 marks max]***

Second Example:

```
public void selectionSortA(int[] numbers)
   {
      int first, current, least, temp;
      for(first = 0; first < 7; first = first + 1)
      {
         least = first;
         for(current = first+1; current < 7; current = current + 1)
         {  if (numbers[current] < numbers[least])
            {  least = current; }
         }
         temp = numbers[least];
         numbers[least] = numbers[first];
         numbers[first] = temp;
      }
   }
```

Award ***[1 mark]*** *for each of the following*:
correct method definition;
array of numbers passed to method;
definition of any variables used;
correct outer loop;
least (or other appropriate) set in correct place;
correct inner loop starting at correct place;
correct test;
correct swap;
no syntax mistakes; ***[8 marks max]***

2. (a) For example:

```
class Student extends Person {
     private int grade;

     public print(){
        if (grade>3)
          {output(name + phone + grade);}
         }
     }
```

Award **[*1 mark*]** *for each of the following*
use of extends to inherit from person;
variables from person not declared;
declare grade;
public print;
correct test and output; ***[4 marks max]***

(b) *Award **[1 mark]** for each of the following*
a record structure;
with three fields;
two pointer fields;
an integer data field; ***[3 marks max]***

(c)
```
class Node
{
   int data;
   TreeNode leftChild;
   TreeNode rightChild;
 }
Node current = root;
public int count=0;
```

/*The above are assumed before the method is called*/

```
public int countNodes(TreeNode current)
     /*
     * function to count nodes in tree
     /*

{
     if (current == null && count ==0)
        {return 0;}
      else {return (1+countNodes(leftChild.TreeNode) +
     countNodes(rightChild.TreeNode));
}
```

Algorithms will vary but award marks as follows:

*Award **[1 mark]** for each of the following*
structure of node clearly declared (can be in part taken from part (b));
first call of method uses the root;
count initialized outside the method;
check for empty tree;

*Award **[2 marks]** for each of the following*
recursive call to left and right;
count accumulated correctly; ***[7 marks max]***

(d) *Award up to **[3 marks]** for a valid comparison of an advantage, **[1 mark]** for an answer with some credit.*

advantages:
the dynamic structure allows growth in size (to the memory limits) whereas the array is fixed/static;
the dynamic structure allows for the data and pointer fields to be of different types whereas all entries into the array must be of one type/integer;

*Award up to **[3 marks]** for a valid comparison of a disadvantage, **[1 mark]** for an answer with some credit.*

disadvantages:
since the array is declared at compile time there is no chance of out of memory errors during program execution;
access to a static structure could be faster/more efficient than access to a dynamic structure (especially with large structures); ***[6 marks max]***

3. (a) *Award **[1 mark]** for each of the following*:
a sum of 9;
intermediate value of 99;
modulo value of 49; ***[3 marks]***
*Award all **[3 marks]** if only this is given*.

(b) 50; ***[1 mark]***

(c) Award ***[1 mark]*** for each problem and ***[1 mark]*** for an elaboration up to 2 x 2 marks maximum, *e.g.*

clashes:
two or more different codes;
yield the same value;

clustering 1:
some letters may be more common than others;
leading to an uneven spread of values;
because A for accounting makes no contribution to the system code (since it equals 0);

clustering 2:
many codes will start with the same letter;
increasing the possibility of clashes/clustering;
because the department letter is first; ***[4 marks max]***

(d) Award ***[1 mark]*** for each problem and ***[1 mark]*** for an elaboration up to 2 x 2 marks maximum, *e.g.*

clashes:
use a linear search;
to locate the next free space;

clustering 1:
alter the calculation to introduce more of a spread;
for example take the square of the middle 1/2 letter(s) and add to the total;
make the code values range from 1 to 26;

clustering 2:
ignore the first letter;
in the calculation; ***[4 marks max]***

(e) *Award **[1 mark**] for each of the following*:
full index will be required;
in which each employee number is entered;
together with the record position in the file;
a linear search could be made using the employee number;
which would be acceptable;
because the file is small; ***[3 marks max]***

(f) *Award **[1 mark]** for each of the following*:
use unicode or ascii values to generate a number;
subtract base value to generate same numbers;
modulo values with a number higher than 50;
use number equal to number of memory locations;
validate unicode or ascii values within correct range;
add difference between lowercase and uppercase base points to low number;
and see if it is lifted into correct range; ***[5 marks max]***

4. (a) *Award **[1 mark]** for each of the following*:
length;
volume;
pitch;
instrument of a note; ***[2 marks max]***

(b) *Award **[2 marks]** for an outline of the way in which each data is used*.
For example: *length and instrument*.
according to instrument the appropriate note;
is sent to the corresponding output channel (instruments have a predetermined channel);
the sound for that instrument is output;
for the length of time indicated; ***[4 marks]***

(c) *batch processing*:
the list of instructions that make up the sound output can be pre recorded or composed from the keyboard and played together at any time to reproduce the music. This is batch processing because once saved together the instructions can be run to produce the music at any time without interference;
[2 marks]

real time processing:
the music can be manipulated as it is being played to produce the required modifications which are reproduced immediately. This is real time because the modifications are made and played as the music is being played; ***[2 marks]***

(d) *Award **[1 mark]** for each of the following*:
sequential;
is only one suitable;
as data is saved in order;
processed in order;
accessed in order; ***[5 marks]***

(e) *Answers to this question must demonstrate an ability to interpret a systems flowchart*.

(i) *input*:
three separate MIDI instruments play and the digital result is sent to the computer sequencer;
via the MIDI IN connection;
as a series of messages; ***[2 marks max]***

process:
the messages are interpreted by the sequencer;
as a series of sounds to be transmitted to separate channels; ***[2 marks]***

output:
each channel transfers data from the computer to the sound system;
and the appropriate sound is output; ***[2 marks]***

(ii) *Award **[1 mark]** for each of the following*:
a Midi Interface card that can cope with multiple inputs and outputs;
high quality sound card;
MIDI sequencer; ***[2 marks max]***

(f) *Award **[1 mark]** for each of the following*:
a driver is a program;
that acts between the computer and another device;
it interprets the instructions from the computer and translates them into a form understandable to the device (or vice versa);
For example: messages from the MIDI sequencer can be translated into the appropriate sounds for the sound device used; ***[4 marks]***

(g) *Award **[1 mark]** for each of the following*.
cyclical series of prototypes;
used to elicit feedback from user;
with increasing accuracy / completeness;
used to match final product with user needs;
and achieve user acceptance; ***[5 marks]***

(h) *Award **[1 mark]** for each of the following*.
a graphics interface;
with simple stop and start buttons to allow the film to be interrupted;
a series of menus to add a section of music;
showing names of music sections;
and fade in – out *etc.*; ***[2 marks max]***

*Award **[1 mark]** for each of the following*.
programmer will discuss with user the needs;
show a prototype;
feed back to complete version;
user test and feedback again until user is satisfied; ***[2 marks max]***

(i) *Award **[2 marks]** for any relevant expanded discussion point up to **two** points*.

For example:

musician's talent to compose acceptable sounds will not change and probably cannot be done by everyone. However, the way in which this is to be recorded will not be the same so the musician will probably have to learn new skills;

instrument players will still be needed to give the input to MIDI systems but once recorded they are no longer needed to play as many different pieces of music can be created from the sounds;

as the system advances high quality sounds may be generated by keyboard (almost there now) and musicians no longer needed to make CDs. However, a live concert will probably always be needed (or not as we become more virtual) because of the atmosphere; ***[4 marks max]***

IB DIPLOMA PROGRAMME
PROGRAMME DU DIPLÔME DU BI
PROGRAMA DEL DIPLOMA DEL BI

SPEC/5/COMSC/SP1/ENG/TZ0/XX

COMPUTER SCIENCE
STANDARD LEVEL
PAPER 1

SPECIMEN

1 hour 30 minutes

INSTRUCTIONS TO CANDIDATES

- Do not open this examination paper until instructed to do so.
- Section A: answer all the questions.
- Section B: answer all questions.

SECTION A

*Answer **all** the questions.*

1. The colour of a pixel can be stored as a 16-bit integer.

(a) State how many different colours can be represented in a 16-bit integer field. *[1 mark]*

(b) State whether this storage system for colour values is digital or analog. *[1 mark]*

(c) Outline **one** advantage and **one** disadvantage of using 32-bits per-pixel to store colours instead of 16-bits per-pixel. *[2 marks]*

2. (a) State the names of **two** standard searching algorithms, including one algorithm which is significantly more efficient than the other. *[2 marks]*

(b) Outline **two** situations where the more efficient searching algorithm cannot be used. *[2 marks]*

3. Documentation for a computer system could be stored as HTML pages and viewed in a web browser.
Outline **two** advantages of storing user documentation in this way, rather than on paper. *[2 marks]*

4. State the output of the following code fragment: *[1 mark]*

```
double n= 1234.5678;
double p = math.floor((n*100)/100);
output (p);
```

Recall that `math.floor(3.7)` produces the integer result 3.

5. Define the term *prototyping*. *[2 marks]*

6. (a) Outline the need for a *data compressor*. *[2 marks]*

(b) Outline **one** example of the use of *data encryption*. *[2 marks]*

7. A robot can be used to transport heavy objects from place to place in a factory.

(a) Outline a situation where the robot needs *voice recognition* capability rather than keyboard input. *[1 mark]*

(b) Describe a standard input system for marking an ID number on the objects, which enables the robot to identify the objects it is transporting. *[2 marks]*

8. (a) Explain the function of *cache memory*, as it relates to the functions of the *CPU*, the *RAM*, and the *data bus*. *[3 marks]*

(b) Outline the need for *virtual memory*. *[3 marks]*

9. Convert the decimal number 97

(a) to a binary number and show how this would be represented using one byte. *[2 marks]*

(b) to hexadecimal. *[2 marks]*

Show all your working out in each case.

SECTION B

*Answer **all** the questions.*

10. A legal office with six lawyers, three secretaries and one receptionist has standalone computers for all those working in the office. It is decided to introduce a local area network. A file server is to be used to hold all files centrally, and all those working in the office will have access via the current computers that they use.

Some of the files, such as the calendar should be available to all users. Others should only be available to one or two of the employees. When legal contracts have been drawn up they should not be changed by anyone.

The software that gives access to the network and allows the user to search through documents on the server is to be written especially for the office.

(a) Outline the advantages of a *prototype* being written before the system is started. *[2 marks]*

(b) Describe the hardware that needs to be installed for the network. You should not include the server and workstations in your answer. *[3 marks]*

(c) Suggest a method that should be employed to maintain *data security*. *[5 marks]*

11. A large library contains approximately ten million books. The library owners wish to have a database system to store the following data for each book. The system should provide online access to the data. Multiple copies of some books are held and each copy is numbered.
A sample data record is shown below:

Title: Java is trivial Author: Theodore Writer ISBN: 0-7506-3241-0 Date: 2001 On loan: yes Copy number: 3 Description :A brief introduction to Java, including sample Applets and a thorough list of web site addresses for downloading source code.

(a) Design a suitable *class definition* (data structure) to store this type of data. *[5 marks]*

During the *systems analysis* phase of development, the problem should be formulated precisely, to ensure that an appropriate solution is produced.

(b) Outline **two** appropriate methods for collecting information from the end-users during this phase. *[2 marks]*

(c) Outline the role of a web browser and a search engine when a person wants to search the Internet for information on a book that is not available in the library. *[3 marks]*

12. A hospital wishes to automate some daily tasks. The following four systems are suggested:

- Payroll – Calculating salaries for employees, including extra pay for overtime work
- Ordering – Ordering supplies and medications when stock levels are low
- Monitoring – Monitoring and recording patients' vital signs: temperature, heart rate, blood pressure, *etc*.
- e-mail – An internal e-mail system for employee communication.

(a) Outline **two** differences in testing the e-mail system and the Monitoring system. *[2 marks]*

(b) Describe an appropriate use of *batch-processing* in one of the four systems. *[2 marks]*

A security subsystem will require users to type a 4 digit integer ID (*e.g*. 1357) before using any computerized device – this includes medical monitoring devices as well as PC terminals. Each user will have their own unique ID number assigned.

(c) (i) Explain one hardware advantage of using an ID number rather than a password containing letters. *[2 marks]*

(ii) A programmer attempted to generate the ID numbers by adding up the ASCII codes of the letters in employees' names. Explain why this method might fail. *[2 marks]*

The ASCII coding system is no longer viable if the system is to be used world wide.

(iii) Identify an alternative system for representing the letters in an employee's name and state why it is needed. *[2 marks]*

13. The sequential file `names` contains a list of names, already sorted in alphabetical order. There are more than 1000 names in the file. A programmer must produce software to detect and remove duplicate names.

(a) State which of the methods below (`aaa` or `bbb`) is better for finding duplicates, and outline **two** advantages it has compared to the other. You may wish to trace the procedures using the sample data shown here:

Arienne
Bill
Bill
Chen

[3 marks]

```
public void aaa ( ) throws IOException
{
     BufferedReader names = new BufferedReader(new FileReader ("names.txt"));
     String na, nb;
     na = names.readLine();
     do
        {nb = names.readLine();
        if (na.equals(nb))
          {output(nb);}
          na = nb;
        }
     while names.ready();
     names.close();
}

public void bbb ( ) throws IOException
{
     BufferedReader names = new BufferedReader (new FileReader ("names.txt"));
     String na, nb;
     do
        {na = names.readLine();
         nb = names.readLine();
     }
     while ((names.ready) &&! (na.equals(nb)));
     if (na.equals(nb))
          {output(nb);}
           names.close();
}
```

(b) (i) Explain why it is sometimes **more efficient** to copy an entire sequential-file into an array before searching for duplicates and then search in the array, rather than searching directly in the file. *[2 marks]*

(ii) Explain why it is sometimes **not possible** to copy a file into an array before searching. *[2 marks]*

(This question continues on the following page)

(Question 13 continued)

(c) Assume the global array `namer` contains 20 names, sorted in alphabetical order and that the following algorithm requires approximately 20 milliseconds to erase all the duplicate names.

```
public void erase ()
{
      int a,b;
      static final int max = 20;
      for (a = 0; a< max; a++)
      { for (b = 0; b< max; b++)
        {if (a != b) && namer[a] = namer[b])
              {namer[b] = ""}
        }
      }
}
```

Predict the time required to use this same algorithm to erase duplicates if the list were twice as long (assuming `max` is changed to the value 40), and explain your calculation. *[3 marks]*

MARKSCHEME

SPECIMEN

COMPUTER SCIENCE

Standard Level

Paper 1

1. (a) 2^{16} (= 65536); ***[1 mark]***

(b) digital; ***[1 mark]***

(c) advantage: many more colours can be held, in fact millions;

disadvantage: takes more space to store / transmit – twice as much; ***[2 marks]***

2. *Accept any sensible answer where one is significantly faster than the other*.
For example:

(a) sequential search and binary search; ***[2 marks]***

(b) *binary search cannot be used when*:
data is not sorted;
data in a sequential file with variable length fields; ***[2 marks]***

3. *Accept any valid advantage* ***[1 mark]*** *for each*.

For example:
is available to the user whilst the computer system is in use and as problems arise;
takes up less space; cannot get lost as easily;
Do not accept both.
user can link quickly to required section and between sections using the hyperlinks; ***[2 marks max]***

4. 1234; ***[1 mark]***

5. *Award* ***[1 mark]*** *for*:
the construction of a simple version of a system in the design stage;

Award ***[1 mark]*** *for either of the following points (or something resembling)*:
showing the user interface but without full processing behind it;
allowing the user to propose changes at the design stage; ***[2 marks max]***

6. (a) *Award* ***[1 mark]*** *for each of the following*.
to shorten the length of files;
by elimination all unnecessary data – empty records, redundancies;
to speed up uploads;
to put on a floppy disk *etc*.; ***[2 marks max]***;

(b) *Award* ***[1 mark]*** *for the application and* ***[1 mark]*** *for elaboration*.
data encryption is used to code data of credit card numbers when transmitted over the Internet so that unauthorised people cannot understand them; ***[2 marks]***

7. (a) when an accident might occur if the robot is not stopped immediately; ***[1 mark]***

(b) bar code printed on object;
and robot fitted with a bar code scanner; ***[2 marks]***

8. (a) *Award **[3 marks]** for any explanation which includes a correct relationship to each of the other components, e.g. **[1 mark]** for CPU, **[1 mark]** for RAM, **[1 mark]** for bus, for example.*
the data bus transfers data from the RAM to the cache;
the CPU retrieves data directly from the cache, without accessing the RAM;
RAM contains a lot of data, but cache only retrieves a small amount at a time; ***[3 marks]***

*It is not necessary to describe technical details such as data-transfer-rates, paging, write-through, etc. A diagram showing the cache sitting between the RAM and CPU should be awarded **[2 marks]** if everything is labelled but no further explanation is provided.*

*Award **[3 marks]** for any explanation which includes a correct relationship.*

(b) *Award **[1 mark]** for each correct point, for example:*
when large programs need to process a lot of data;
it is not always possible for them to reside in RAM;
hence part of disk is simulated as RAM;
so that program can run;
efficiency (speed) affected,

[3 marks max]

9. (a) 64 32 16 8 4 2 1 ;
1 1 0 0 0 0 1 which means 01100001 to fit in one byte; ***[2 marks]***
*Award **[1 mark]** only for 1100001.*

(b) 0110 = 6 0001=1 ;
61 in hexadecimal ; ***[2 marks]***
Allow follow through.

10. (a) *Award **[1 mark]** for each of the following and any other different and valid point.*
a prototype allows the user to have more input/change mind about what is needed;
programmer can incorporate these changes into the system;
likely that final system will suit requirements better; ***[2 marks max]***

(b) *Award **[2 marks]** for the following.*
hub or switch to receive/send data to and from the server and workstations;

*Award **[1 mark]** for each of the following.*
cables that connect each machine to the hub;
and connect the hub to the server; ***[3 marks max]***
*Do not award **[2 marks]** for hub or switch without some elaboration as to why they need to be installed.*

(c) *Award **[1 mark]** for each of the following.*
Data could be saved in separate areas (folders) on the server;
separate access rights given to different groups;
so that some parts had restricted access and could only be read and written to by one or two employees and some parts available to everyone;

*Award **[2 marks]** for the following.*
once legal documents completed they should be automatically transferred (by the software) to an area which is read only (or the document properties changed to read only by the software) on confirmation of completion;

*Award **[2 marks]** for the following.*
could also archive off to CD so that if tampered with and changes made there would be a dated version to compare with; ***[5 marks max]***

11. (a)

```
public class Book
{ public static void main (String[] args)
     {new Book();}
     public Book()
     { String title, author, ISBN;
          int year;
          boolean loan;
          int copy;
          String description;
     }
}
```

*Award **[1 mark]** for each of the following.*
title, author and description as `String`;
ISBN as `String`;
year as integer;
loan as boolean;
copy as integer; ***[5 marks]***

(b) *Award **[1 mark]** for each of the following.*
interviews with all users;
and questionnaires handed out to all users; ***[2 marks]***

(c) *Award **[1 mark]** each for the following points* (*maximum of **[2 marks]** for correct function of browser and maximum of **[2 marks]** for correct function of search engine)*

user selects suitable search engine;
the web browser will display the page that gives links to searching;
user will input keywords for search;
search engine searches database of keywords to find match;
browser displays the page that gives results to search; ***[3 marks max]***

12. (a) *Award **[1 mark]** for any statement which gives a legitimate difference and which gets over the idea that there can be no possibility of failure for the monitoring.*

For example:
testing the monitoring system cannot be done with real patients in case there is a failure;
testing the monitoring system will have to test all possible paths whereas email needs only to check a random number of send and receives; ***[2 marks]***

(b) payroll since on a regular monthly basis the same program will be run to generate the payslips; ***[2 marks]***
*Award **[1 mark]** only for payroll alone as a description is asked for.*

(c) (i) the ID number can be held as long integer which takes less space than letters also is quicker to compare; ***[2 marks]***

(ii) unlikely to give unique codes;
because two people could have the same letters in a different order / different letters could add to the same; ***[2 marks]***

(iii) unicode would have to be used instead of ASCII since only the extended Latin alphabet is available in ASCII and other script (*e.g.* Chinese) would need more letters available; ***[2 marks]***

13. (a) `aaa` is better;
it continues to the end and finds all duplicates (unlike bbb which stops after the first);
`bbb` will only find those, which occur as pairs in 1st and 2nd, 3rd and 4th etc. not those in 2nd and 3rd, 4th and 5th *etc.*; ***[3 marks]***

(b) (i) reading a whole block is more efficient than reading smaller sections many times (transfer time) would also be more efficient if the deletion was to be done when duplicates found since no shuffling or flagging would be needed; ***[2 marks]***

(ii) file may be too large to fit into RAM or it could be of unknown size and array could not be dimensioned in advance; ***[2 marks]***

(c) loop carried out 20 x 20 times in 20 milliseconds;
if carried out 40 x 40 times this would take 4 times longer;
hence 80 milliseconds; ***[3 marks]***

IB DIPLOMA PROGRAMME
PROGRAMME DU DIPLÔME DU BI
PROGRAMA DEL DIPLOMA DEL BI

COMPUTER SCIENCE
STANDARD LEVEL
PAPER 2

SPECIMEN

1 hour 30 minutes

INSTRUCTIONS TO CANDIDATES

- Do not open this examination paper until instructed to do so.
- Answer all questions.

1. In an online computer game, the player's name is entered at the start of each game and at the end of the game a number of points is recorded according to the skill shown in the game. A Hall of Fame holds the names and top scores of the fifty best players, in order of score. At the end of each game the number of points gained is checked against the scores in the Hall of Fame. If it is greater than the lowest score, that name and score are removed from the list and the current player's name and score are inserted. A class `Hall` has been defined such that the names of the players in the Hall of Fame are held in the array `player` and their corresponding scores in the array `score`. Assume that the arrays are full.

(a) Construct an algorithm for a Boolean method that checks to see if a player qualifies for the Hall of Fame. Name the method `check(points)` where `points` is the number of points that the player has at the end of the game. *[4 marks]*

(b) State why a Boolean method is appropriate for the algorithm. *[2 marks]*

A new player plays the game and qualifies. The name and points have to be added to the Hall of Fame. One method to do this is to search the array `score` sequentially from the first element to find the position where the player should be inserted. When this place is found the player's name and score are inserted, and all those below are moved down one position.

(c) Construct an algorithm for a method `place` that accepts the name and points of a qualifying player and inserts these in the Hall of Fame. *[6 marks]*

No player appears twice in the Hall of Fame so if a player who qualifies is already listed then only the better score is entered.

(d) Construct an algorithm for a method `update` to check whether a qualifying player is already in the list and if so either replace the old score with the new one or leave the previous score. Remember that the Hall of Fame must still be in order of scores. *[8 marks]*

2. The police station in a certain town analyses the different types of motoring offences committed each month. The class structure for the data sent is as follows.

```
public class Crimedata
{ public static void main (String[] args)
  { new Crimedata();}
   public Crimedata()
   { int month; //month number//
     int crime1,crime2,crime3,crime4;
//total number of each crime committed//
    }
}
```

The data for one year is held in a two dimensional array.

(a) Create a new array `yearData` to hold the crime figures for one year in a two dimensional array. You do not need to hold the month number. This array will replace the integers declared in the above code. *[2 marks]*

(b) Add the necessary statements to read all the crime figures for one year into your array. *[3 marks]*

(c) Write the code for a method `maxCrime` which can be incorporated in the `CrimeData` class and which takes in a crime number as parameter and outputs the month in which the maximum number of that crime occurred, together with the actual number of crimes. For example, if the maximum number of `crime4` is 75 and these occurred in February, `CrimeData.maxCrime(4)` would result in an output of:

2 75. *[8 marks]*

(d) Write a method for the `CrimeData` class which will create a list (or one dimensional array) of the average number of each crime committed within the year. For example, the first value in the list will be the average number of crime1 committed over the twelve months. Call the list `averageCrimeList` and explain where and how it is declared. *[7 marks]*

3. *This question requires the use of the Case Study*.

A phrase of music to simulate being played by an orchestra can be reduced to a set of instructions.

(a) Identify **two** items of data that would be processed by the instructions. *[2 marks]*

(b) Outline how these **two** items would be used in the execution of an instruction. *[4 marks]*

(c) Explain the use of *batch processing* and *real time processing* in the use of MIDI devices. *[4 marks]*

(d) With reference to the system flowcharts (figure 6 on page 10 and figure 8 on page 11)

(i) Outline the *input – processing – output* that takes place when more than one MIDI instrument is used to produce a complex musical composition. *[6 marks]*

(ii) Identify **two** features of a home computer that make it suitable to produce professional quality music from input via the keyboard *[2 marks]*

(e) Describe the role of *drivers* in the control of hardware devices, using MIDI as an example. *[4 marks]*

A software writer has been approached to create a program that film makers can use to add music to films. The person employed to do this is called a mixer. The musician has composed suitable music for the film and the mixer is required to add appropriate parts at the right time in the film. The mixer is extremely good at choosing the right passages of music to add to a film and knows when they need to be faded in and out, but she is not a computer expert.

(f) Suggest a suitable user interface for the mixer and explain how this can be tested out by the program writer. *[4 marks]*

(g) Discuss the implications to professional musicians of the use of MIDI in the creation and playing of music. *[4 marks]*

MARKSCHEME

SPECIMEN

COMPUTER SCIENCE

Standard Level

Paper 2

1. (a) Assume the array `score` is already declared as an array of integer. The given algorithms are examples only. Solutions may vary and still be correct.

```
public void boolean check(int points)
  {if points>score[49]
     {return true; }
     else
   {return false; }
  }
```

*Award **[1 mark]** for each of the following.*
points sent as integer parameter;
correct method declaration;
correct test points;
correct return; ***[4 marks]***

(b) because it must return a value;
which can only be true or false; ***[2 marks]***

(c)

```
public void place(int points, String name)
  {int count = 0;
    while points <score[count]
   {count = count+1; }
     if count != 49
       {for( int i = count; i < 50; i = i + 1)
          {player[i + 1] = player[i];
           score[i + 1] = score[i]; }
        }
     player[count] = name;
     score[count] = points;
     }
   }
```

*Award **[1 mark]** for each of the following.*
intialise count;
correct loop to find position;
check not at end of list;
shuffle down players;
shuffle down scores;
place player in correct place;
place score in correct place; ***[6 marks max]***

(d)

```
public void Update(String name, int points)
    {int count =0;
       boolean replace = false;
       boolean found = false;
       do
      {if name ==player[count]
           {found = true;
              if score[count] < points
           {replace = true;}
             else
           {count=count+1;}
            }
      }
       while !found && count <50
       if found && replace
      {if count < 50
          {for (int i=count; count<49;count=count+1)
                {player[i]=player[i+1];
                  score[i]=score[i+1];
                 }
           }
      }
         Hall.place(points,name);
     }
```

*Award **[1 mark]** for each of the following.*
correct parameters;
initialisation of all variables;
loop until player found;
test to see if scorer is greater;
if present and to be replaced;
delete player from previous position;
shuffle the rest;
call place(points,name) correctly; ***[8 marks]***

2. (a)

```
int[][] yearData = new int[12][4] ;
```

Award 1 mark for each of the following. The order of 12 and 4 does not matter
New Object declared as integer
Correct dimensions ***[2 marks]***

(b) *For example.*

```
for (int i=0; i<12;i=i+1)
  {for (int j=0; j<4;j=j+1)
     {yearData[i][j] = inputInt("crime figures");}
   }
```

*Award **[1 mark]** for each of the following.*
double loop;
correct order for i and j (according to part (a)) and correct limits;
correct input statement; ***[3 marks]***

(c) *For example.*

```
public void maxCrime(int crime)
{int max=0;
 int maxMonth=0;
   for (int i=0; i<12;i=i+1)
      {if yearData[i][crime]>max
         {max=yearData[i][crime];
               maxMonth=i+1;
              }
       }
      output("maximum number of crimes" +crime + max);
      output ("was in month" + maxMonth);
  }
```

*Award **[1 mark]** for each of the following.*
parameter crime as integer;
max and maxMonth declared;
max initialised;
correct loop;
correct test;
max assigned;
maxMonth assigned;
correct output; ***[8 marks]***

(d)

```
double averageCrimeList[4]                (1)
//already declared outside the method          (1)

void calculateAverages()
      {for (int i=0; i<4;i=i+1)
            int tot=0
          {for (int j=0; j<12;j=j+1)
            tot=tot+yearData[j][i];
            }
            averageCrimeList[i]=tot/12
      }
```

*Award **[1 mark]** for each of the following points.*
correct declaration of averageCrimeList;
as double;
outside the method;
for each crime in turn;
initialise tot or equivalent;
accumulate over each month;
assign average to list;
dividing by 12; ***[7 marks max]***

3. (a) *Award **[1 mark]** for each of the following.*
length;
volume;
pitch;
instrument of a note; ***[2 marks max]***

(b) *Award **[2 marks]** for an outline of the way in which each data is used.*
For example: *length and instrument*.
according to instrument the appropriate note;
is sent to the corresponding output channel (instruments have a predetermined channel);
the sound for that instrument is output;
for the length of time indicated; ***[4 marks]***

(c) *batch processing*:
the list of instructions that make up the sound output can be pre recorded or composed from the keyboard and played together at any time to reproduce the music. This is batch processing because once saved together the instructions can be run to produce the music at any time without interference;
[2 marks]

real time processing:
the music can be manipulated as it is being played to produce the required modifications which are reproduced immediately. This is real time because the modifications are made and played as the music is being played; ***[2 marks]***

(d) *Answers to this question must demonstrate an ability to interpret a systems flowchart.*

(i) *input*:
three separate MIDI instruments play and the digital result is sent to the computer sequencer;
via the MIDI IN connection;
as a series of messages; ***[2 marks max]***

process:
the messages are interpreted by the sequencer;
as a series of sounds to be transmitted to separate channels; ***[2 marks]***

output:
each channel transfers data from the computer to the sound system;
and the appropriate sound is output; ***[2 marks]***

(ii) *Award **[1 mark]** for each of the following.*
a Midi Interface card that can cope with multiple inputs and outputs;
high quality sound card;
MIDI sequencer; ***[2 marks max]***

(e) *Award **[1 mark]** for each of the following.*
a driver is a program;
that acts between the computer and another device;
it interprets the instructions from the computer and translates them into a form understandable to the device (or vice versa) ;
For example: messages from the MIDI sequencer can be translated into the appropriate sounds for the sound device used; ***[4 marks]***

(f) *Award **[1 mark]** for each of the following.*
a graphics interface;
with simple stop and start buttons to allow the film to be interrupted;
a series of menus to add a section of music;
showing names of music sections;
and fade in – out *etc.*; ***[2 marks max]***

*Award **[1 mark]** for each of the following.*
programmer will discuss with user the needs;
show a prototype;
feed back to complete version;
user test and feedback again until user is satisfied; ***[2 marks max]***

(g) *Award **[2 marks]** for any relevant expanded discussion point up to **two** points.*

For example.

musician's talent to compose acceptable sounds will not change and probably cannot be done by everyone. However, the way in which this is to be recorded will not be the same so the musician will probably have to learn new skills;

instrument players will still be needed to give the input to MIDI systems but once recorded they are no longer needed to play as many different pieces of music can be created from the sounds;

as the system advances high quality sounds may be generated by keyboard (almost there now) and musicians no longer needed to make CDs. However, a live concert will probably always be needed (or not as we become more virtual) because of the atmosphere; ***[4 marks max]***

IB DIPLOMA PROGRAMME
PROGRAMME DU DIPLÔME DU BI
PROGRAMA DEL DIPLOMA DEL BI

SPEC/5/COMSC/BP2/ENG/TZ0/XX/CS

COMPUTER SCIENCE
CASE STUDY

SPECIMEN

INSTRUCTIONS TO CANDIDATES

- Case study booklet required for higher level paper 2 and standard level paper 2 computer science examinations.

MIDI – Musical Instrument Digital Interface.

MIDI is the use of computer programs and sound cards to play numerous musical parts with realistic sounds.

History

Before MIDI was invented a composer had to get a whole band together in order to hear what a composition sounded like. This was time-consuming, expensive and difficult to arrange. But it had to be done after every re-write in order to hear the latest version. MIDI makes it possible to hear a piece of music without the need for musicians who are capable of learning to play the music fast. MIDI makes it possible for one person to:

- Create music that would require many musicians to play.
- Engineer difficult passages.
- Play the music.
- Edit the music.
- Post the music.
- Create music even if one can't play an instrument.

MIDI requires fewer people, less equipment and no studio facilities. Instead of hiring musicians, equipment and studios to record a composition after every change, the composer can record each track into the computer whenever it is convenient. Once in the computer the music can be edited often until it is perfect. Corrections can be made to individual notes so that new versions of the music do not require the musicians to replay all the 'good' sections again. Difficult sections can be played in at a slow speed, or even engineered, and replayed at full speed without affecting the sound quality. Additionally, the whole piece can be edited if needed. For example, every note could be raised by a semitone at the press of a button. This would normally require the whole piece to be re-recorded. Replays can be heard immediately after each edit and multiple versions can be saved for comparison. This leaves the composer free to compose music instead of spending large amounts of time organizing recording sessions.

The cost of MIDI technology is falling and performance is rising. So much so that very high quality equipment is now available to the general public. By comparison, the cost of using a studio will keep pace with wage, rent and equipment costs. MIDI technology is, in effect, a "Band in a Box" that is available 24 hours a day, 7 days a week. When the composer has an inspirational idea, the band is there, ready and waiting to show the composer what the idea will actually sound like. The idea is also saved for future reference.

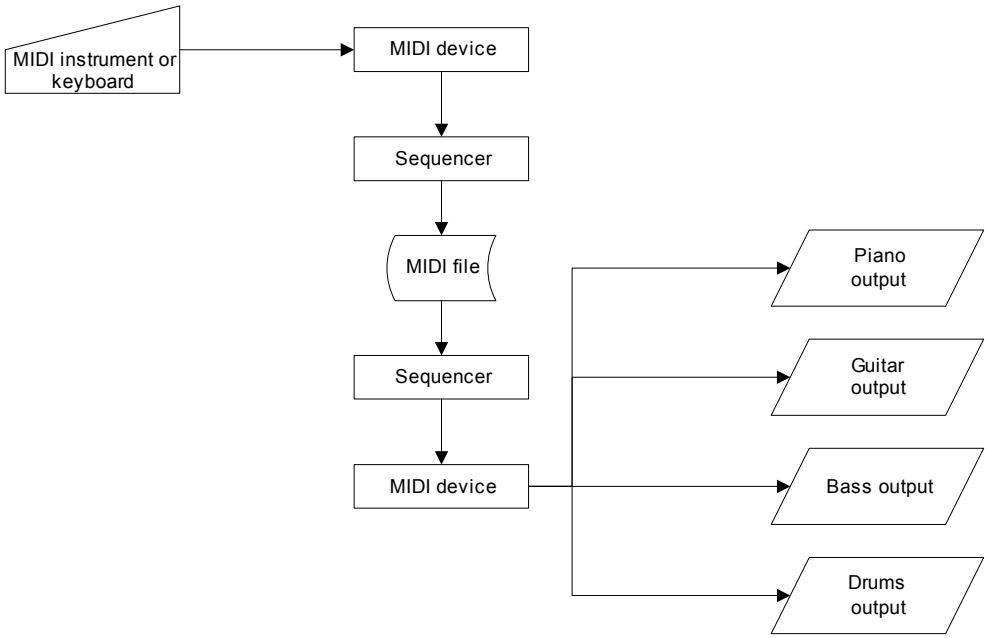
Input to a MIDI file is usually via a MIDI keyboard but other instruments can be modified to generate MIDI data. A guitar, for example, usually generates an analog signal but, fitted with the right sensors, it can generate digital MIDI data. This same data can be used to drive a sound module that is set up to produce the sound of a harp, or a flute, or a piano. In fact anything, the sound of a frog croaking can be used as the sound source and suddenly the frog is singing the composer's tune. If you can play one MIDI instrument you can produce the sound of any real instrument and more - in effect creating music that is impossible using conventional instruments.

MIDI emerged alongside synthesisers. Musicians wanted to control them with computers and at the same time one of the first home computers, the Commodore 64, became available to the public. The Commodore had a built-in analog synthesiser chip and musicians began to use it. The PC had no built in chip but it wasn't long before a slot-in card with a built in chip was available. It came with a language suitable for transferring data, MIDI, which was adopted by many manufacturers and the musical revolution had begun. Software, called a sequencer, was developed to make controlling the chip easy, MIDI interfaces were added to many devices and the MIDI Manufacturers Association was created to establish and control industry wide protocols.

Computerised control allows the musician to:

- Control electronic instruments remotely.
- Control electronic instruments automatically.
- Control more than one instrument.
- Combine sounds.
- Use one sound module to create several instruments.
- Create music that is physically impossible to play.

The instructions for controlling MIDI devices can be batch processed (a pre-recorded list of instructions) or processed in real time (instructions generated during performance).



How it works

When a musician uses a conventional instrument to create a note, it always starts with an action such as blowing, plucking, pressing, bowing or hitting. This is called a 'NOTE-ON' event in MIDI. The sound can be stopped sometime later when the musician ceases to execute the 'NOTE-ON' event. Stopping the sound in this way is a 'NOTE-OFF' event. The sound will also have a 'PITCH' corresponding to the underlying frequency of the sound wave. A high 'PITCH' note such as that produced by a whistle will have a high frequency.

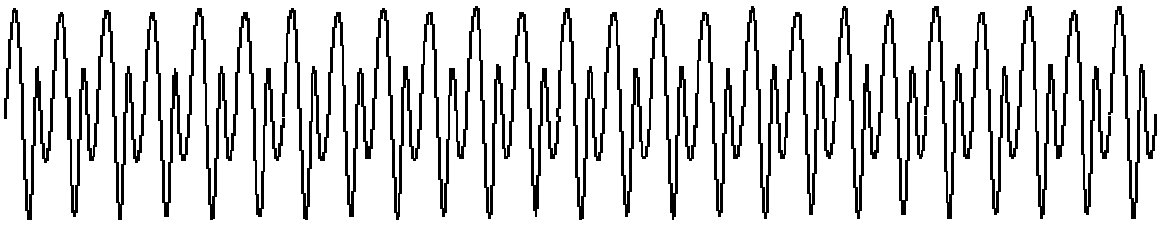


Figure 1: High Frequency

A low ‘PITCH’ note such as that produced by a rumble will have a low frequency.

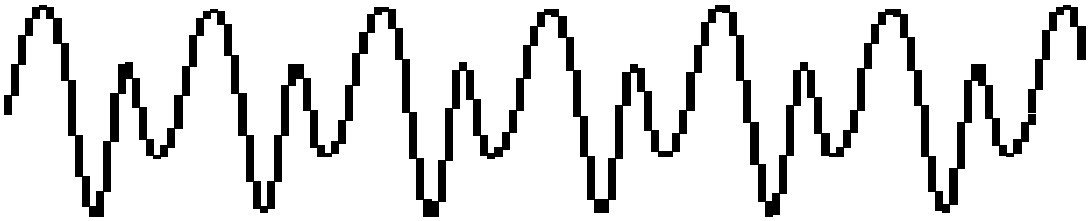


Figure 2: Low Frequency

The sound will also have a ‘VOLUME’ or amplitude depending on how loud it is. A quiet sound will have a low ‘VOLUME’ and a loud sound will have a high ‘VOLUME’. The speed at which the note builds up from zero volume to full volume is called the ‘ATTACK’. Something that is hit, such as a drum, reaches full volume almost straight away and is said to have a fast ‘ATTACK’. The gentle build up volume that can be produced by bowing a cello has a slow ‘ATTACK’. The speed at which a note fades away is called the ‘DECAY’. A drum beat fades away quickly and so has a fast ‘DECAY’, whereas a piano note, so long as the key remains pressed, loses volume slowly and so has a slow ‘DECAY’. There are other PARAMETERS that can be applied to a note but these are the main ones and collectively they define the ENVELOPE of the note.

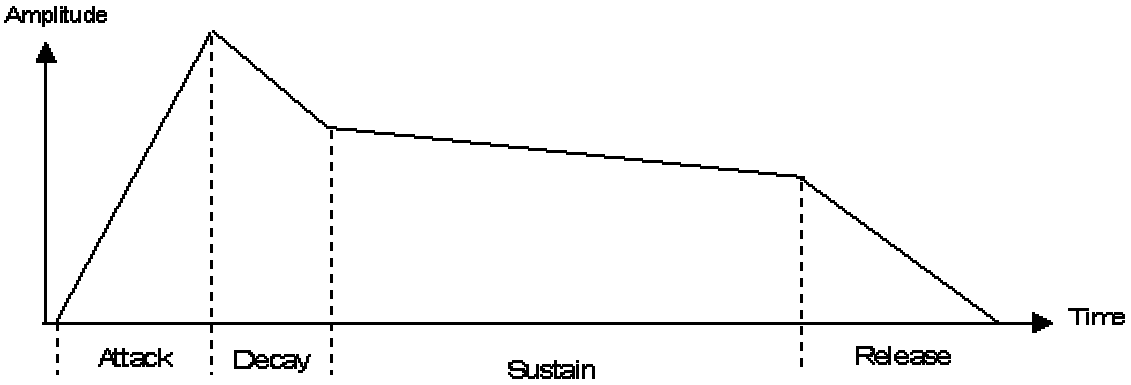


Figure 3: Parameters of a note. Courtesy of MIDI Manufactures Association – Used with permission.

When the envelope parameters are combined with a soundwave the result is a note.

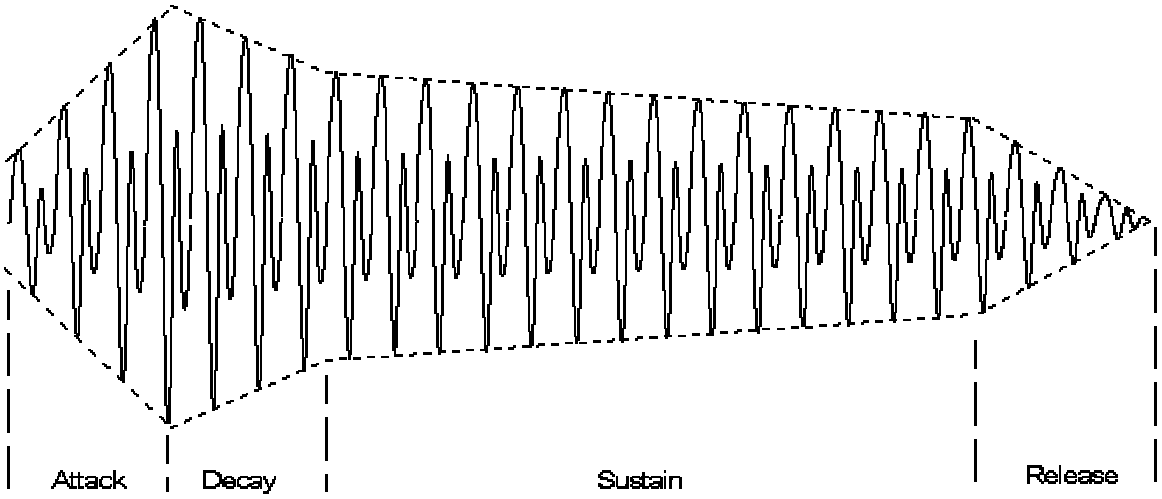


Figure 4: A sound wave modified by an envelope. Courtesy of MIDI Manufactures Association – Used with permission.

A musical note can therefore be stored as a list of integers representing all of its parameters. Storing the data this way makes it easy to manipulate. For example, to increase the volume of a note just increase the value of the Volume parameter. Or to change the note itself just change the Pitch value. It also produces very small files compared with other methods of recording. This makes them very suitable for use in small devices and for transmission. For example a tune can easily be stored on a small chip such as those found in greeting cards or delivered as part of an on-line computer game. And because it's all just numbers it is easy to handle digitally without the attenuation affects suffered by analog signals.

The sound wave inside the envelope can be created in two ways. Frequency Modulation (FM) synthesis and Wavetable synthesis.

With FM synthesis, a minimum of two periodic signals are mixed together and their parameters adjusted to create a sound. These sounds are not the sounds of real instruments but can be made to sound similar. FM synthesis is better for 'electronically' created sounds.

If real instrument sounds are needed then digital sampling systems are used to store high quality sound samples digitally, and then replay them as needed. Because samples require large amounts of memory, looping, pitch shifting, mathematical interpolation, and digital filtering can be used to reduce the amount of sampling required. This type of synthesis is called "wavetable" synthesis and can be thought of as a "table" of sounds which may be looked up and used. One-Shot Sounds such as short drum sounds are stored as a single sample which is played once through with no processing to save memory.

Data Compression can be used to improve the signal-to-noise ratio for some samples. The dynamic range of the sound samples is reduced and then decompressed during playback to restore the dynamic range. There are several techniques which may be used to compress the dynamic range of a signal.

Because the data for each instrument is delivered on a different track it can be transmitted between devices on different channels and every aspect of the sound can be easily edited and other parameters easily added. Not only can performance parameters be added (‘slides’, ‘slurs’, ‘pull offs’ etc) but colouration (‘echo’, ‘phase’, ‘sustain’ etc) can be applied.

In comparison, a digitised recording of an analog instrument has to be sampled about 44,000 times per second. This means the amplitude of the sound wave is measured 44,000 times each second and the value stored as a number. That’s 44,000 numbers being saved for each second of performance, and that has to be done for each instrument. It is also more difficult to edit afterwards. If instruments have been mixed to produce a final performance it is very difficult to pull them apart again and the results may be unpredictable. This makes it more demanding on resources and takes control away from the composer.

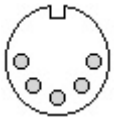
For example, if you play a note on a keyboard, and hold it for 1 minute, the NOTE-ON and NOTE-OFF MIDI messages will be stored in 6 bytes. To do the same thing with a digital recording will take 44,000 * 60 samples because all of the sound between the start of the note and the end of the note has to be recorded. MIDI records the musician’s actions, not the sound so it just has to save the events. The spaces between events are not recorded. MIDI takes about 10 Kbytes per minute of performance to store and digital audio takes about 10 Mbytes per minute of performance to store.

The major disadvantage of MIDI is that it cannot handle the singing of words. When the voice is used as an instrument (‘oohs’ and ‘aahs’) a sample of the sound can be made and played back as a parameter in a MIDI message. But because the words of a song are changing continuously they cannot be stored as a MIDI message parameter.

Communication

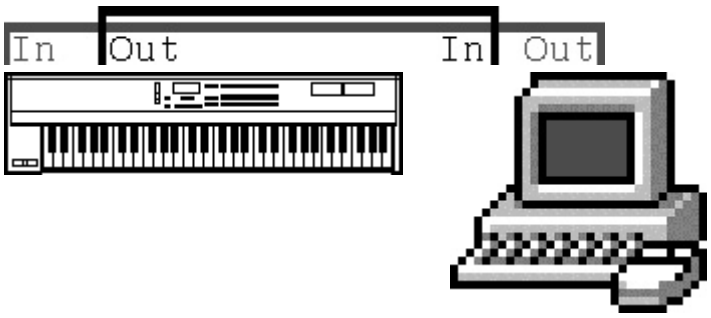
MIDI data flows in one direction (unidirectional) and does not handshake the target device before transmission (asynchronous). This means that the target device must be dedicated to the source and be available at all times to receive data. Data flows at a rate of 31.25 Kbits/s and is transmitted as 10-bit bytes: a start bit, 8 data bits and a stop bit. Signals are generated by MIDI controllers or sequencers.

MIDI controllers are devices that are attached to musical instruments that generate MIDI data as the musician is playing the instrument (real time processing). They are electronic circuits connected to keys, pick ups or microphones. A MIDI sequencer is a device which allows MIDI data to be captured, stored, edited and replayed (batch processing). A sequencer can be a computer application or a dedicated hardware device that can do everything a software version can do but it does it with electronic circuits and not computer program instructions. The hardware version is therefore faster and of a higher quality but the software version is easier to update or upgrade. The messages are transmitted via a MIDI OUT connection and received by a MIDI IN connection. There will often be a MIDI THRU connection available for messages that need to be passed from machine to machine without being altered in any way.



The connectors themselves are called 5 pin DIN jacks and separate connectors are used for IN, OUT and THRU signals. The internal wiring between pins means that general purpose DIN-to-DIN leads cannot be used and special MIDI leads have to be used. They look the same but the internal pin-to-pin connections are different. When a computer is connected to a keyboard MIDI connection, the keyboard is used to create performance data, the computer is used to store, edit and send it back to the keyboard. When the keyboard re-receives the MIDI data, it uses its onboard sound chips to reproduce the sounds.

Figure 5:



A single message sent between machines will represent a single instruction such as a NOTE-ON command. A typical message will often be 3 bytes in length: a status byte and 2 data bytes, For example, if a musician presses middle c on a keyboard the midi circuits inside the keyboard will send the numbers 144 60 64 to the MIDI OUT connectors. 144 means NOTE-ON. 60 means middle c. 64 is the velocity of the key press. When the key is released the numbers 128 60 64 will be sent. If the MIDI IN connector of a second device were connected to the same MIDI OUT connector, the same note would be played. If the first device were set to a piano sound, and the second device set to an organ sound, the result would be a piano and an organ playing the same note. Linking devices together like this is called ‘Daisy Chaining’. Communications are serial and, when many machines are linked together, many events may need to be triggered at the same time. On the first note of a bar, for example, 10 instruments all playing a 5 note chord will generate 50 NOTE-ON messages needing to be delivered at the same time. Because the system is serial the 50 events will have to be triggered one after another. However, as it takes about 1ms to deliver a MIDI message, all 50 events can be triggered in 50 ms. To the human ear they sound simultaneously. Where delays may be audible a feature called ‘running status’ can be used to combine similar events in a single message. Messages will be either system or channel (voice and mode) messages.

Channel messages can be of type:

- Note On,
- Note Off,
- Velocity,
- Aftertouch
- Sustain,
- Key Pressure,
- Channel Pressure,
- Pitch Bend,
- Program Change.

System messages can be of type:

- Bank Select,
- RPN / NRPN,
- Channel Mode Messages,
- System Messages,
- System Common Messages,
- System Real Time Messages,
- System Exclusive Messages.

MIDI data can be stored in three formats. Format 0 stores all of the MIDI sequence data in a single track. This makes the data file small and uses less bandwidth for transmission but it is difficult to edit because all the parts are mixed. Format 1 files store MIDI data as a collection of tracks. The files are bigger than format 0 but they are easy to edit as all the different channels have been preserved. Format 2 files can store several channels using independent patterns. Format 1 is the most often used.

Messages can be transmitted along 16 channels and devices can be set to respond to a pre-selected channel. Middle c, NOTE-ON transmitted on channel 1 will be 144 60 64. The same note on channel 2 will be 145 60 64. Four bits within the message bit stream select the channel. This means that data sent around a daisy chain can be accepted or ignored by devices on the transmission path depending on which channels they are set to receive data. In effect, the devices can all be playing different notes using the data being transmitted around the system.

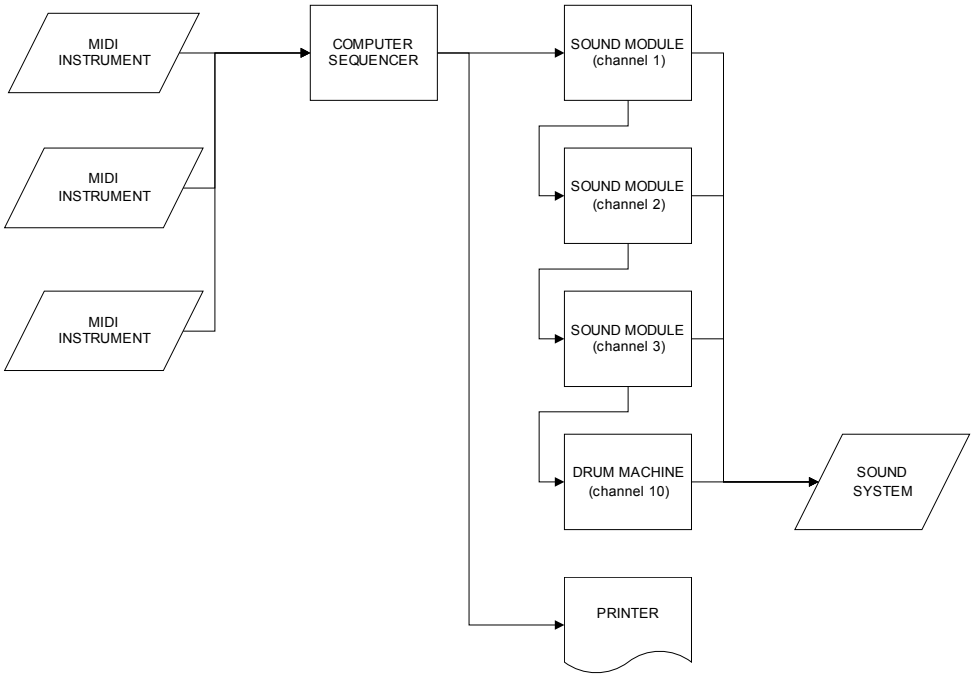


Figure 6:
Courtesy of MIDI Manufactures Association – Used with permission

The sounds produced by each sound module will depend on what sound patches have been assigned to each channel. The General MIDI system utilises channel 10 for drums and all the other channels for chromatic instruments. Instrument sounds are grouped into sets. For example, program numbers 1-8 are piano sounds, 9-16 are tuned percussion sounds, 17-24 are organ sounds, 25-32 are guitar sounds, etc. There are 128 sounds available. Channel 10 is therefore different from the rest as each note will produce the sound of a different instrument (bass drum, cymbal, clap) rather than the same sound produced at a different pitch.

Although three different input devices are shown above, often only one is used: a keyboard. The computer could be a small dedicated sequencer. And if the sound module were a type that could produce different sounds (polyphonic multi-timbral) on different channels at the same time then the system would look like:

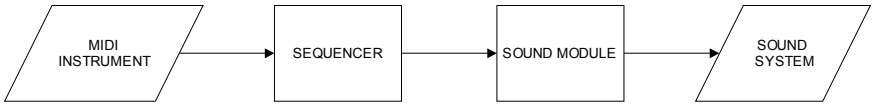


Figure 7:
Courtesy of MIDI Manufactures Association – Used with permission

Even a rudimentary system like this can reproduce the sound of an entire band. The sound card in a PC has been of limited quality in the past but better quality cards can be added, making a PC-based home recording studio look the following:

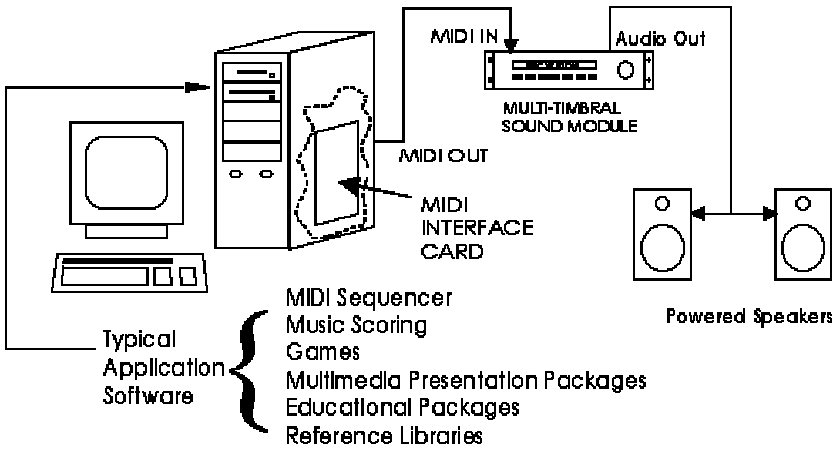


Figure 8: **A PC-Based MIDI System**. Courtesy of MIDI Manufactures Association – Used with permission.

Configuration

Computer applications address hardware devices through the use of drivers. These provide applications software with a common interface to access hardware simply. When a MIDI interface or synthesiser is installed in a computer a suitable device driver has to be loaded. In Windows a MIDI Mapper applet is used to receive messages from an application, which then sends the messages to the device driver.

Conclusion

The storage efficiency and ease of editing makes MIDI an attractive tool for generation of sounds in multimedia applications, computer games, portable devices or high-end karaoke equipment. It also opens up its own avenues for musical expression to the musically literate and illiterate alike. One such application uses the electronic systems found in burglar alarms to generate the MIDI messages. Proximity sensors are used to detect movement within a performance space. The speed of movement and positioning is used to generate pitch and volume data. Computers are then used to normalise this data to a set of rules and to generate other signals that are used to create backing. Rather than the dancer having to follow the music, the dancer can thus create to music and never be out of step.