Multimedia BSc Exam 2000 SOLUTIONS

Setter: ADM
CheckerACJ

Additional Material: SMIL Language Description Sheet

Answer 3 Questions out of 4

1. (a) *What is meant by the terms Multimedia and Hypermedia? Distinguish between these two concepts.*

Multimedia ---- An Application which uses a collection of multiple media sources e.g. text, graphics, images, sound/audio, animation and/or video.

Hypermedia --- An application which uses associative relationships among information contained within multiple media data for the purpose of facilitating access to, and manipulation of, the information encapsulated by the data.

2 MARKS ---- BOOKWORK

(b) What is meant by the terms static media and dynamic media? Give two examples of each type of media.

Static Media – does not change over time, e.g. text, graphics

Dynamic Media --- Time dependent (Temporal), e.g. Video, sound, animation.

4 MARKS --- BOOKWORK

(c) What issues of functionality need to be provided in order to effectively use a wide variety of media in Multimedia applications? Your answer should briefly address how such functionality can facilitated in general Multimedia applications.

The following functionality should be provided:

-   Digital Representation of Media --- Many formats for many media
-   Capture: Digitisation of Media --- special Hardware/Software
-   Creation and editing --- assemble media and alter it
-   Storage Requirements --- significant for multimedia
-   Compression --- related to above and below, ie can save on storage but can hinder retrieval
-   Structuring and retrieval methods of media --- simple to advanced DataBase Storage

- Display or Playback methods --- effect of retrieval must view data
- Media Synchronisation --- display multimedia as it is intended
9 MARKS --- BOOKWORK

(d) Different types of media will require different types of supporting operations to provide adequate levels of functionality. For the examples of static and dynamic media given in your answer to part 1(b) briefly discuss what operations are need to support a wide range of multimedia applications.

A selection of the items below is reuired for good marks NOT ALL. Other Solns Possible?

Typical Range of operations required for common media

Text: Editing
          Formatting
          Sorting
          Indexing
          Searching
          Encrypting
          ABOVE REQUIRE: :
                    Character Manipulation
                    String Manipulation

Audio: Audio Editing
              Synchronisation
              Conversion/Translation
              Filtering/ Sound Enhancing Operators
              Compression
              Searching
              Indexing
          ABOVE REQUIRE: :
                    Sample Manipulation
                    Waveform Manipulation

Graphics: Graphic primitive Editing
               Shading
               Mapping
               Lighting
               Viewing
               Rendering
               Searching
               Indexing
               ABOVE REQUIRE: :
               Primitive Manipulation
               Structural/Group Manipulation

Image: Pixel operations
          Geometric Operations
          Filtering
          Conversion
          Indexing
          Compression
          Searching

Animation: Primitive/Group Editing
          Structural Editing
          Rendering
          Synchronistaion
          Searching
          Indexing

Video: Pixel Operations
          Frame Operations
          Editing
          Synchronisation
          Conversion
          Mixing
          Indexing
          Searching
          Video Effects/Filtering

12 MARKS --- UNSEEN

2. *(a) Why is file or data compression necessary for Multimedia activities?*

   Multimedia files are very large therefore for storage, file transfer etc. file sizes need to be reduced. Text and other files may also be encoded/compressed for email and other applications.

2 MARKS --- BOOKWORK

(b) *Briefly explain, clearly identifying the differences between them, how entropy coding and transform coding techniques work for data compression. Illustrate your answer with a simple example of each type.*

Compression can be categorised in two broad ways:

      Lossless Compression
        -- where data is compressed and can be reconstituted (uncompressed) without loss of detail or information. These are referred to as bit-preserving
        or reversible compression systems also.
      Lossy Compression
        -- where the aim is to obtain the best possible fidelity for a given bit-rate or minimizing the bit-rate to achieve a given fidelity measure. Video and
        audio compression techniques are most suited to this form of compression.

      Lossless compression frequently involves some form of entropy encoding and are based in information theoretic techniques

      Lossy compression use source encoding techniques that may involve transform encoding, differential encoding or vector quantisation.

ENTROPY METHODS:

The entropy of an information source S is defined as:

$H(S) = SUM_I (P_I Log_2 (1/P_I)$

      where P_I is the probability that symbol S_I in S will occur.

 $Log_2 (1/P_I)$ indicates the amount of information contained in S_I, i.e., the number of bits needed to code S_I.

Encoding for the Shannon-Fano Algorithm:

A top-down approach

1. Sort symbols according to their frequencies/probabilities, e.g., ABCDE.

2. Recursively divide into two parts, each with approx. same number of counts.

(Huffman algorithm also valid indicated below)

A simple transform coding example

A Simple Transform Encoding procedure maybe described by the following steps for a 2x2 block of monochrome pixels:

1. Take top left pixel as the base value for the block, pixel A.
2. Calculate three other transformed values by taking the difference between these (respective) pixels and pixel A, i.e. B-A, C-A, D-A.
3. Store the base pixel and the differences as the values of the transform.
Given the above we can easily for the forward transform:

and the inverse transform is trivial

The above transform scheme may be used to compress data by exploiting redundancy in the data:

Any Redundancy in the data has been transformed to values, Xi. So We can compress the data by using fewer bits to represent the differences. I.e if we use 8 bits per pixel then the 2x2 block uses 32 bits/ If we keep 8 bits for the base pixel, X0, and assign 4 bits for each difference then we only use 20 bits.
Which is better than an average 5 bits/pixel

8 MARKS --- BOOKWORK

(c) (*i*) *Show how you would use Huffman coding to encode the following set of tokens:*

BABACACADADABBCBABEBEDDABEEEBB

How is this message transmitted when encoded?

The Huffman algorithm is now briefly summarised:

1. Initialization: Put all nodes in an OPEN list, keep it sorted at all times (e.g., ABCDE).

2. Repeat until the OPEN list has only one node left:

      (a) From OPEN pick two nodes having the lowest frequencies/probabilities, create a parent node of them.
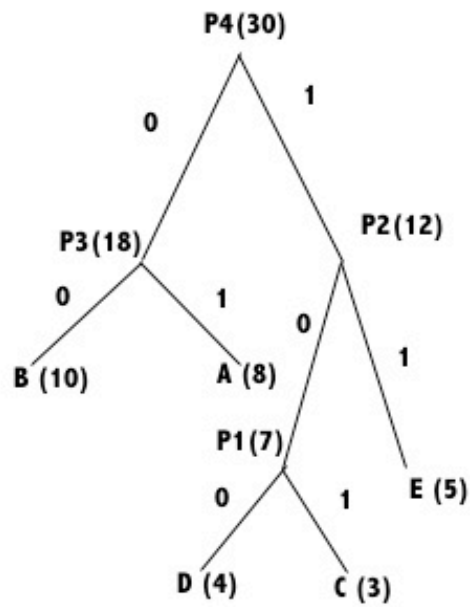
      (b) Assign the sum of the children's frequencies/probabilities to the parent node and insert it into OPEN.

      (c) Assign code 0, 1 to the two branches of the tree, and delete the children from OPEN.

Symbol	Count	OPEN (1)	OPEN (2)	OPEN (3)	OPEN (4)
A	8			18	30
B	10			-	
C	3	7	12		-
D	4	-			
E	5		-		
Total	30				

-   indicate merge node with other node with number in column

Finished Huffman
Tr

Symbol	Code
A	01
B	00
C	101
D	100
E	11

How is this message transmitted when encoded?

Send code book and then bit code for each symbol.

7 Marks --- UNSEEN

(ii) How many bits are needed transfer this coded message and what is its Entropy?

Symbol	Count	Subtotal # of bits
A	8	16
B	10	20
C	3	9
D	4	12
E	5	10

Total Number bits (excluding code book) = 67

Entropy = 67/30 = 2.2333

4   MARKS --- UNSEEN

(iii)   What amendments are required to this coding technique if data is generated live or is otherwise not wholly available? Show how you could use this modified scheme by adding the tokens ADADA to the above message.

Adaptive method needed:

Basic idea (encoding)

```
Initialize_model();
 while ((c = getc (input)) != eof)
    {
      encode (c, output);
      update_model (c);
    }
```

So encode message as before:

A= 01 D = 100

So addd stream:

011000110001

Modify Tree:

Symbol	Count	OPEN (1)	OPEN (2)	OPEN (3)	OPEN (4)
A	11			21	35
B	10			-	
C	3	8	14		-
D	6		-		
E	5	-			

Huffman tree drawn as before but different.

6 Marks --- UNSEEN

3 (a) What are the major factors to be taken into account when considering what storage requirements are necessary for Multimedia Systems?

Major factors:
    Large volume of date
    Real time delivery
    Data format
    Storage Medium
    Retrieval mechanisms

4 MARKS --- Unseen/applied bookwork

(b) What is RAID technology and what advantages does it offer as a medium for the storage and delivery of large data?

RAID --- Redundant Array of Inexpensive Disks

Offers:
    Affordable alternative to mass storage
    High throughput and reliability

RAID System:
    Set of disk drives viewed by user as one or more logical drives
    Data may be distributed across drives
    Redundancy added in order to allow for disk failure

4 MARKS --- BOOKWORK

(c) Briefly explain the eight levels of RAID functionality .

Level 0 – Disk Striping --- distributing data across multiple drives
Level 1 – Disk Mirroring --- Fault tolerancing
Level 2 – Bit Interleaving and HEC Parity
Level 3 - Bit Interleaving with XOR Parity
Level 4 – Block Interleaving with XOR Parity
Level 5 - Block Interleaving with Parity Distribution
Level 6 – Fault Tolerant System --- Error recovery
Level 7 – Heterogeneuos System --- Fast access across whole system

8 MARKS --- BOOKWORK

(d) A digital video file is 40 Mb in size. The disk subsystem has four drives and the controller is designed to support read and write onto each drive, concurrently. The digital video stored using the disk striping concept. A block size of 8 Kb is used for each I/O operation.

(i) What is the performance improvement in sequentially   reading the complete file when compared to a single drive subsystem in terms of the number of operations performed?

We have 5120 segments to write to RAID disks. Given 4 disks we have 1280 actual I/Os to perform

On 1 drive we clearly have 5120 operations to perform.

(ii) What is the percentage performance improvement expressed as the number of physical I/O operations to be executed in on the RAID and single drive systems?

The improvement is
(5120 – 1280)/1280*100 = 300%. Obvious given 4 concurrent drives and RAID!!

11 MARKS --- UNSEEN

4 (a) Give a definition of a Multimedia Authoring System. What key features should such a system provide?

An Authoring System is a program which has pre-programmed elements for the development of interactive multimedia software titles.
Authoring systems vary widely in orientation, capabilities, and learning curve.

There is no such thing (at this time) as a completely point-and-click automated authoring system; some knowledge of heuristic thinking and algorithm design is necessary.

Authoring is basically just a speeded-up form of programming --- VISUAL PROGRAMMING; you don't need to know the intricacies of a programming language, or worse, an API, but you do need to understand how programs work.

2 MARKS ---- BOOKWORK

(b) *What Multimedia Authoring paradigms exist? Describe each paradigm briefly.*

There are various paradigms, including:

Scripting Language

The Scripting paradigm is the authoring method closest in form to traditional programming. The paradigm is that of a programming language, which specifies (by filename) multimedia elements, sequencing, hotspots, synchronization, etc. A powerful, object-oriented scripting language is usually the centerpiece of such a system; in-program editing of elements (still graphics, video, audio, etc.) tends to be minimal or non-existent. Scripting languages do vary; check out how much the language is object-based or object-oriented. The scripting paradigm tends to be longer in development time (it takes longer to code an individual interaction), but generally more powerful interactivity is possible. Since most Scripting languages are interpreted, instead of compiled, the runtime speed gains over other authoring methods are minimal.

The media handling can vary widely; check out your system with your contributing package formats carefully. The Apple's HyperTalk for HyperCard, Assymetrix's

OpenScript for ToolBook and Lingo scripting language of Macromedia Director are examples of a Multimedia scripting language.

Here is an example lingo script to jump to a frame

```
global gNavSprite
on exitFrame
```

```
go the frame
play sprite gNavSprite
end
```

Iconic/Flow Control

This tends to be the speediest (in development time) authoring style; it is best suited for rapid prototyping and short-development time projects. Many of these tools are also optimized for developing Computer-Based Training (CBT). The core of the paradigm is the Icon Palette, containing the possible functions/interactions of a program, and the Flow Line, which shows the actual links between the icons. These programs tend to be the slowest runtimes, because each interaction carries with it all of its possible permutations; the higher end packages, such as Authorware or IconAuthor, are extremely powerful and suffer least from runtime speed problems.

Frame

The Frame paradigm is similar to the Iconic/Flow Control paradigm in that it usually incorporates an icon palette; however, the links drawn between icons are conceptual and do not always represent the actual flow of the program. This is a very fast development system, but requires a good auto-debugging function, as it is visually un-debuggable. The best of these have bundled compiled-language scripting, such as Quest (whose scripting language is C) or Apple Media Kit.

Card/Scripting

The Card/Scripting paradigm provides a great deal of power (via the incorporated scripting language) but suffers from the index-card structure. It is excellently suitedfor Hypertext applications, and supremely suited for navigation intensive (a la Cyan's "MYST" game) applications. Such programs are easily extensible via XCMDs andDLLs; they are widely used for shareware applications. The best applications allow all objects (including individual graphic elements) to be scripted; many entertainment applications are prototyped in a card/scripting system prior to compiled-language coding.

Cast/Score/Scripting

The Cast/Score/Scripting paradigm uses a music score as its primary authoring metaphor; the synchronous elements are shown in various horizontal tracks withsimultaneity shown via the vertical columns. The true power of this metaphor lies in the ability to script the behavior of each of the cast members. The most popularmember of this paradigm is Director, which is used in the creation of many commercial applications. These programs are best suited for animation-intensive orsynchronized media applications; they are easily extensible to handle other functions (such as hypertext) via XOBJs, XCMDs, and DLLs.

Macromedia Director uses this .

Hierarchical Object

The Hierarchical Object paradigm uses a object metaphor (like OOP) which is visually represented by embedded objects and iconic properties. Although the learning curve is non-trivial, the visual representation of objects can make very complicated constructions possible.

Hypermedia Linkage

The Hypermedia Linkage paradigm is similar to the Frame paradigm in that it shows conceptual links between elements; however, it lacks the Frame paradigm's visual linkage metaphor.

Tagging

The Tagging paradigm uses tags in text files (for instance, SGML/HTML, SMIL (Synchronised Media Integration Language), VRML, 3DML and WinHelp) to link pages, provide interactivity and integrate multimedia elements.

8 Marks --- BOOKWORK

(c)   You have been asked to provide a Multimedia presentation that can support media in both English and French. You may assume that you have been given a sequence of 10 images and a single 50 second digitised audio soundtrack in both languages. Each Image should be mapped over consecutive 5 second fragments of the audio. All Images are of the same 500x500 pixel dimension.

Describe, giving suitable code fragments, how you would assemble such a presentation using SMIL. Your solution should cover all aspects of the SMIL presentation

```
<smil>
        <head>
        <layout>
         <root-layout height="500" width="500" background-
color="#000000" title="MultiLingual"/>
          <region id="image1" width="500" height="500" top="0"
left="0" background-color="#000000" z-index="1" />
          <region id="image2" width="500" height="500" top="0"
left="0" background-color="#000000" z-index="1" />
………..
        </layout>
        </head>

<body>
     <par>
        <switch>

        <!-- English only -->
```

```
                < audio system-language="en" src ="english.au" />

                  <!-- French only -->
                  <audio system-language="fr" src ="francais.au" />
              </switch>

<seq>
                  <img src="image1.jpg" region="image1" begin="0.00s"
dur="5.00s" />
                  <img src="image2.jpg" region="image2" begin="5.00s"
dur="5.00s" />

…….

</seq>
</par>

</body>
</smil>
```

17 Marks ---- UNSEEN