StudentId______________

VICTORIA UNIVERSITY OF WELLINGTON
Te Whare Wananga o te Upoko o te Ika a Maui

EXAMINATIONS — 2006

END-YEAR

COMP 206

PROGRAM AND DATA STRUCTURES

Time Allowed: 3 Hours (180 minutes)

Instructions:

- Attempt all questions.
- There are 180 possible marks on the exam.
- Make sure your answers are clear and to the point.
- Non-programmable calculators without full alphabetic keys are permitted.
- Non-electronic foreign language dictionaries are permitted.
- Refer to the Appendix.
- No other reference material is allowed.
- Answer in the appropriate heavily outlined boxes or follow the instructions given in the questions.

Question	**Topic**	**Marks**
PART 3		
10	File Structure Fundamentals	[25 marks]
11	B-tree	[20 marks]
12	Index-Sequential File	[15 marks]
13	Secondary Indices	[10 marks]
14	Hash File	[10 marks]

StudentId______________

Question 10. File Structure Fundamentals [25 marks]

a) [2 marks] Define the file record format.

ANSWER

The record format is a named sequence of fields containing (field name, data type) pairs. It is defined using struct or the private part of a class.

b) [2 marks] Define the record key**.**

ANSWER

The record key is a sequence of record format fields whose composite value uniquely identifies each record in a file. A key should be non redundant. Each key value has to be defined.

c) [9 marks] Describe each of the three basic file organizations using the following two criteria:

- The way file records are assigned to storage locations, and
- The relationship between a record's key value and the relative address of the location the record is stored in.

ANSWER

The heap file:
Records are stored densely in successive location according to the order of their entry and regardless to their key value.

The sequential file:
Records are stored densely in successive locations. A record with a greater key value occupies a location with a greater relative address.

The direct file:
A record is stored in the location whose relative address is a function of the record's key value. There may be non occupied locations in the file.

d) [12 marks] Suppose the declarations of a struct and a variable given below are defined in a program.

```
typedef struct {
     int StudentId;
     char Name[16];
     char Address[31];
     } Record;
     …
     record student;//student variable
```

Suppose there is the following command

```
FILE *sptr = fopen(student.data, w);
```

and it returnes a not null `sptr` value.

I. [3 marks] Suppose the `student.data` file contains records of a predictable length. Use the `fprintf` C Stream function to write a student record into the `student.data` file.

ANSWER

```
int retv = fprintf(sptr, “%10d %15s %30s”, student.sid,
student.name);
```

II. [9 marks] Suppose the `student.data` file contains records with length indicators in front of each record and each field. Write a part of a C program that will compute the actual length of each field and the record itself and then use the `fprintf` C Stream function to write a student record into the `student.data` file.

ANSWER

```
char intToStr[6]; //Needed for casting int into str
short idlength = strlen(sprintf(intToStr, “%i”,
student.sid));
short nlength = strlen(student.name);
short alength = strlen(student.address);
short length = strlen(buffer);
fprintf(sptr, “%d ”, length); //write record length
fprintf(ptr, “%2d %d%2d %s%2d %s”,
idlength, student.sid, nlength, student.name,
alength, student.address);//write record
```

StudentId______________

SPARE PAGE FOR EXTRA ANSWERS

Cross out the rough working that you do not want marked.
Specify the question number for work you do want marked.

StudentId_______________

Question 11. B-tree [20 marks]

a) [8 marks] In a B-tree of the order *p* = *2m* + *1* and the height *h*:

I. [2 marks] What is the minimum number of (key, address) pairs in a node that is not the root?

ANSWER

m

II. [2 marks] What is the maximum number of (key, address) pairs in a node?

ANSWER

2m

III. [2 marks] What is the minimum number of (key, address) pairs in the root node?

ANSWER

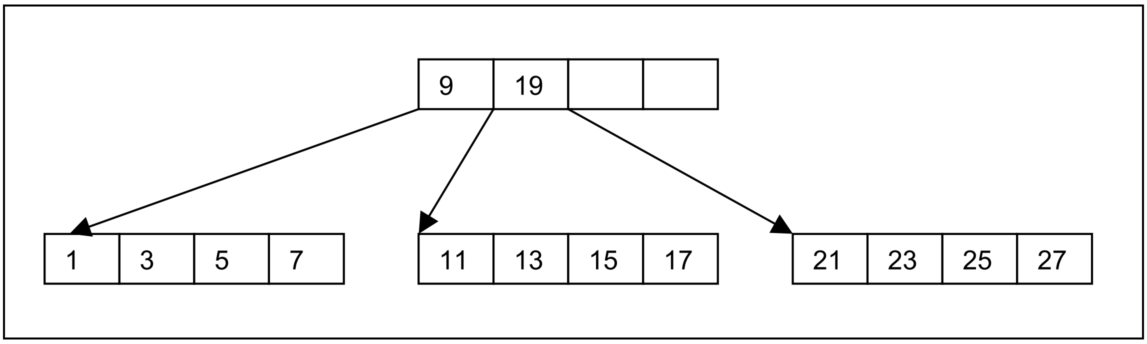
1

IV. [2 marks] What is the number of edges between the root and a leaf node expressed in terms of the height *h*?

ANSWER

h - *1*

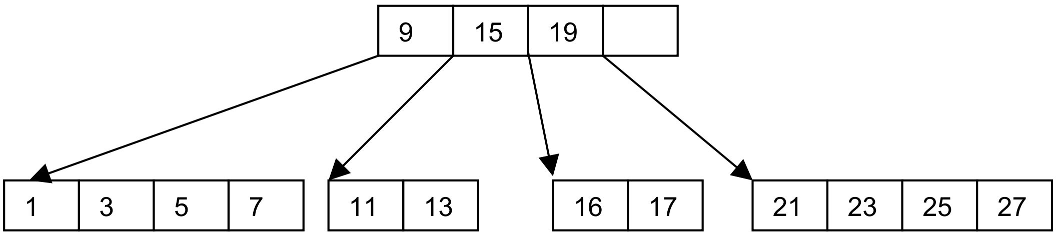
b) [7 marks] Consider the B-tree of the order 5 in Figure below. The address components of node entries are omitted for the sake of simplicity.



Update the B-tree by inserting the key value 16. In your answer, show the B-tree after inserting.

StudentId______________

ANSWER



c) [5 marks] The Admin Node of a B-tree file similar to one you have seen in the Assignment 3 contains the following fields:

- `int num_records // number of records in the file,`
- `int num_of_nodes // number of actual tree nodes,`
- `int num_of_blocks // number of blocks allocated so far to the file,`
- `int ROOT // the relative address of the root node.`

The variable `node_size` contains the length of a node. The file is implemented as a binary file.

Suppose a node splits. How does the `btree.cpp` program compute the relative address of the new node?

ANSWER

The relative address of the new node is

```
(num_of_blocks + 1)*node_size
```

StudentId______________

Question 12. Index-Sequential File [15 marks]

The file header of an index-sequential file with a B-tree is stored in a file allocation table in the main memory. The file header contains various information about the file like: number of blocks allocated to the file, the address of the B-tree root node, the address of the left most sequence set, and the number of records in the file. The file contains $r = 65000$ records. File records have a fixed size of $L = 300$ bytes. File blocks have a size of $B = 4096$ bytes. Each block has a header of $d = 96$ *bytes*.

a) [3 marks] Calculate the range of values of the number s of sequence sets.

ANSWER

Blocking factor $f = \lfloor (B - d)/L \rfloor = \lfloor 4000/300 \rfloor = 13$

$\lceil r/f \rceil \leq s \leq \lceil 2r/f \rceil$, $5000 \leq s \leq 10000$

b) [12 marks] The file is processed sequentially. The average access time to a sequence set (contained in a block on disc) is *3 ms*, the time to read a block into the main memory is *2 ms*, and the time to process a sequence set is *4 ms*.

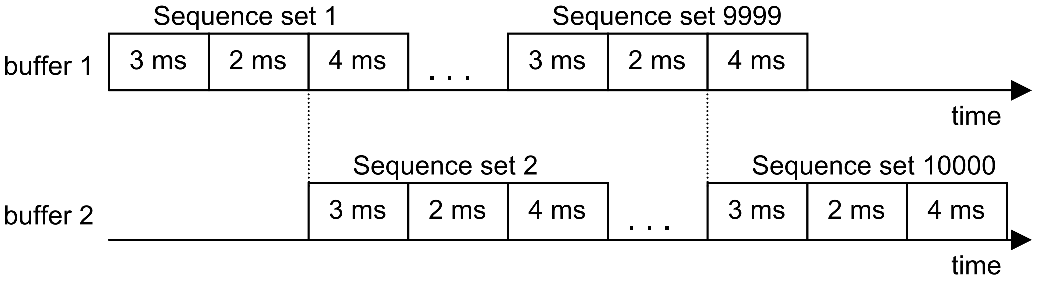
I. [3 marks] Suppose there is only one buffer of *4096* bytes allocated to the index-sequential file. Calculate the expected time to process the file in the worst case.

ANSWER

$t = 10^4 * 9 * 10^{-3} = 90\ s$

II. [9 marks] Suppose there are two buffers of *4096* bytes allocated to the index-sequential file. Calculate the expected time to process the file in the worst case.

ANSWER



$t = 9999*5*10^{-3} + 9*10^{-3} \approx 50\ s$

Question 13. Secondary Indices [10 marks]

Suppose:

- An *Exam* file contains *r = 90000* records,
- Record format is *Exam*(*int StudentId, char CourseId, char Term, char Grade*),
- The file is stored on disk and its structure consists of a data area and several indices,
- There is a secondary index on *Grade* containing *10* secondary key entries,
- There is a secondary index on *CourseId* having *200* secondary key entries,
- There is a secondary index on *Term* having *10* secondary key entries,
- All pointers are *p* = *8* bytes long,
- The file block size is *B* = *8192* bytes, and each block contains a pointer to the next level of indirection.
- All distributions are even, and
- There are several records containing *Grade* = "*A+*", or *CourseId* = "*COMP206*", or *Term* = "*2006T2*" in the file.

How many accesses to disk will it be needed to evaluate the query

> *Retrieve all exam records having Grade ="A+" AND CourseId = "COMP206" AND Term= "2006T2".*

ANSWER

[All lines 1.5 marks]

Number of records with *Grade ="A+"* is 90000/10 = 9000 [1.5 mark]

So, the *Grades* index has $\lceil 9000*8/(8192 - 8) \rceil$ = 9 levels of indirection

Number of records with *CourseId* = "*COMP206*" is 90000/200 = 450

So, the *CourseId* index has $\lceil 450*8/8184 \rceil$ = 1 level of indirection

Number of records with *Term ="2006T2"* is 90000/10 = 9000

So, the *Term* index has $\lceil 9000*8/8184 \rceil$ = 9 levels of indirection

The number of records satisfying all three conditions is

$$\lceil 90000/10*200*10 \rceil = 5$$

The number of accesses is

$$1 + 9 + 1 + 1 + 1 + 9 + 5 = 27$$

Question 14. Hash File [10 marks]

a) [2 marks] What is a hash function?

ANSWER

A hash function is a mapping from a set of record keys into a set of file relative addresses.

b) [2 marks] What are synonyms?

ANSWER

Synonyms are two records with different key values that map into the same file relative address.

c) [2 marks] What is a bucket?

ANSWER

A bucket is a storage place (usually a block) for storing a number (usually greater than 1) of synonyms.

d) [2 marks] What is the home bucket of a record?

ANSWER

The home bucket of a record is the bucket where the hash function maps the record.

e) [2 marks] What is an overflow record?

ANSWER

An overflow record is a record that can't be stored in its home bucket because it is already full.

****

APPENDIX

Low Level I/O System Calls:

```
int fd = open(const char file_name,
                              int flags, [mode_t pmode]);
```

- `flags (O_RDWR | O_RDONLY | O_WRONLY, [O_CREATE], [O_APPEND], [O_TRUNC],...)`

```
ssize_t retval = write(fd, source, size);

ssize_t retval = read(fd, dest, size);

off_t seekval = lseek(int fd,
                            off_t offset, int reference);
```

- `reference – SEEK_SET | SEEK_CUR | SEEK_END`

C Stream File I/O Commands (Text File)

```
FILE *sptr = fopen(char file_name, char file_type);
int fprintf(FILE *sptr, control_string, arg1,…, argn);
int fscanf(sptr, control_string, arg1,…, argn)
       – control string – formatting information,
       – argi (1 ≤ i ≤ n) - individual output data items
long seekval = fseek(FILE *spr, long offset, int ref);
```

- `ref – 0 for SEEK_SET, 1 for SEEK_CUR, 2 for SEEK_END`

File Performance Formulae:

blocking factor $f = \lfloor (B - header)/L \rfloor$
number of blocks $b = \lceil r/f \rceil$
external sort-merge $N = 2b(1 + \lceil (log_{n-1}b) - 1 \rceil)$
number of buffers n

B-tree (the worst case)

$h = 1 + \lfloor log_{m+1}((r + 1)/2) \rfloor$
number of leaves = $2(m + 1)^{h-2}$

B$^+$-tree (the worst case)

$h = 2 + \lfloor log_{m+1}(r/2m) \rfloor$
number of leaves = r/m

Index-Sequential File with a B-tree

number of sequence sets s
$\lceil r/f \rceil \le s \le \lceil 2r/f \rceil$