1                                                                      (10%).
(a) A computer whose processes have 2048 pages in their address spaces keeps in its page tables in memory. The overhead required for reading a word from the page table is 200 nsec. To reduce this overhead, the computer has an associative memory, which holds 32 pairs, and can do a look up in 80 nsec. What hit rate is needed to reduce the mean overhead to 120 nsec?

(b) What is the relationship between the size of the associative memory and locality?

2. As shown in the following
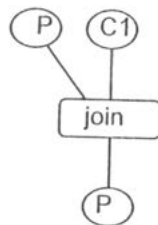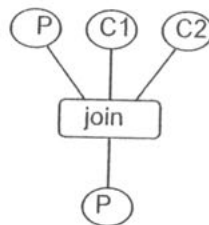


(a)                                    (b)

where, P is parent process and C are children' processes. Explain how to achieve the operations of (a) and (b) in terms of UNIX system call. (Jsut describe the concept, you don't need to write complete code)

                                                                       (10%)

3.
(a) Make the following code complete                                   (10%)
(b) How to use the code for critical-section problem

```
wait_signal: monitor;
        begin
            busy : boolean;
            free : condition;
            procedure mwait;
                begin
                        if busy then free;
                        busy :=
                end;

            procedure msignal;
                begin
                        busy :=;
                        free
                end;
            {initialization}
            busy := false
        end
```

4. (a) A machine has 48-bit virtual addresses and 32-bit physical addresses. Pages are 8K. How many entries are needed for a conventional page table? For an inverted page table? (5%)

(b) Discuss whether each of the following programming techniques and program actions is good or bad with regard to the degree of locality of reference it is likely to exhibit. Explain your reasoning.

(1) sequential processing of a one-dimensional array
(2) sequential processing of a two-dimensional array
(3) Hashing
(4) indirect addressing (10%)

5. Suppose a file system uses the following definitions for its internal structures:

#define BLOCKSIZE 1024

```
struct inode
{
    unsigned long file_size; /* number of bytes in file */
    unsigned long direct[12]; /* direct_zone mappings */
    unsigned long single[1]; /* single_indirect mappings*/
    unsigned long double[1]; /* double_indirect mappings*/
}
```

Assume that zones are exactly one block in size, and that an *unsigned long* number requires 4 bytes of storage.

(a) What is the maximum possible size for a given file ? (using an arithmetic expression to show the answer) (5%)

(b)
Suppose a user-level program opens a file, seeks to some arbitrary position *pos* within the file, and then invokes the *read* system call to read data from this position. The file system code must determine the number of the zone in which the desired data resides. Write C code to implement the following function, which determines the zone number associated with a seek position of *pos* for the given file: (10%)

unsigned long findzone (struct inode *inode, unsigned long pos)

6. The shortest job first (SJF) algorithm is the optimal CPU scheduling. How about the shortest-seek-time-first (SSTF) for disk scheduling in terms of the tracks that the schedulings across ? (i.e. is SSTF optimal ? why?) (10%)

7. Compare the differences between a compiler and an interpreter in terms of running speed, memory space required, and debugging facility. (10%)

8. Draw a figure to show the difference between a linkage editor and a linking loader. (10%)

9. For the following grammar: E → E+E | id , (10%)
(a) Is the grammar ambiguous? (if your answer is NO, skip the (b) question)
(b) If yes, please draw two possible parse trees.