

**Wednesday 9 December 1998**

---

**1. List those items for which memory space is allocated by each of the following methods. (12 points)**

**a. Static memory allocation**

Program code (from text segment), initialized global variables (from data segment), uninitialized global variables (from bss segment)

**b. Dynamic memory allocation**

Heap (for use by new and malloc), stack (for passing parameters to procedures and for local variable storage)

**2. It is possible for a single instruction to page fault more than one time? Explain your answer. (8 points)**

Yes. It can page fault on the instruction fetch, and again on the operand access. If the instruction supports more than one memory operand (for example, a CISC 3-operand arithmetic operation), then it can page fault on each operand fetch. Furthermore, with indirect addressing, the instruction can page fault once when fetching the pointer and again when fetching the operand that the pointer points to.

**3. Compare the following four methods of memory management in a multiprogrammed operating systems, with respect to (i) how much of the program must be loaded into memory, (ii) any constraints on where the program must be loaded in memory, and (iii) what kind of fragmentation is possible and why. (30 points)**

**a. Fixed-size partitions**

Entire program must be loaded.

Program can be loaded into any partition of sufficient size.

Internal fragmentation is possible if the process doesn't fill the entire partition.

**b. Segmentation**

Entire program must be loaded.

Each segment of the program can be loaded into any free memory space of sufficient size; segments need not be loaded contiguously into memory.

Name: \_\_\_\_\_

External fragmentation is possible between segments, if the resulting memory space isn't big enough for other segments to use.

**c. Paging**

Entire program must be loaded.

Each page of the program can be loaded into any free memory frame (all are of the same size); pages need not be loaded contiguously into memory.

Internal fragmentation is possible in the last frame of a process if the process doesn't fill the entire frame.

**d. Demand paging**

Pages of the program are loaded only as needed.

Each page of the program can be loaded into any free memory frame (all are of the same size); pages need not be loaded contiguously into memory.

Internal fragmentation is possible in the last frame of a process if the process doesn't fill the entire frame.

**4. List several features of each of the following types of networks. (10 points)**

**a. Wide area networks**

Large geographic area, bandwidth is scarce, less reliable, high error rate

**b. Metropolitan area networks**

Interconnects LANs, spans city or country, owned by multiple organizations

**c. Local area networks**

Small geographic area, high bandwidth, more reliable, low error rate

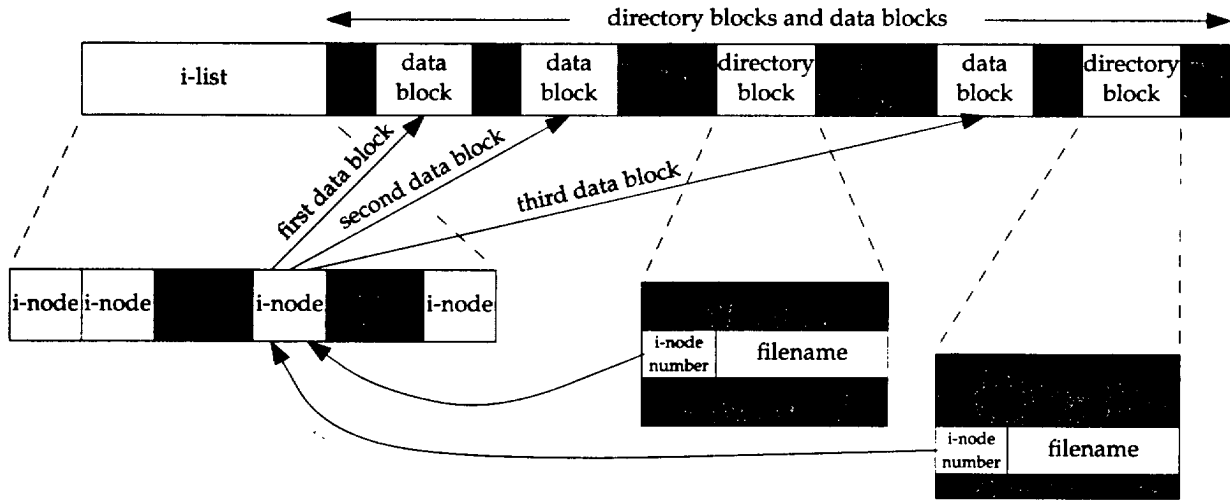
**5. List those items stored in a typical file descriptor. (10 points)**

Type, access permissions, link count, owner and group, size, access times, etc.

Blocks where file is located on disk

(Not stored — name of the file)

**6. Draw a diagram showing the contents of a UNIX disk partition, including at least one file and one directory. (15 points)**



**7. In the changes that were made to UNIX, resulting in Berkeley BSD 4.2 UNIX, (15 points)**

**a. Why was it desirable to increase the size of the disk blocks?**

The main reason was to increase the data transfer rate (after one seek and rotational delay, more data could be retrieved from the disk block).

Other benefits were: (i) to increase the amount of usable disk space (proportional to space used for headers), and (ii) to decrease the size of the free block bit map.

**b. Why was it desirable to not increase the size of the disk blocks?**

Most files are small, so large disk blocks can waste a lot of space due to internal fragmentation.

**c. How was this conflict resolved?**

In two ways: (i) by introducing "cylinder groups" to keep data localized and reduce seek time, and (ii) by increasing the size of the disk blocks but allowing each to be divided into smaller "fragments" as necessary.