Solutions for the First Quiz

COSC 6360
Fall 2017
First question

- How many lines will be printed by the following program?

```c
#include <stdio.h>

main() {
    printf("Start\n");
    fork();
    fork();
    fork();
    printf("Done!\n");
}
```
Answer

- How many lines will be printed by the following program?

- `#include <stdio.h>
  main()
  {
    printf("Start\n");
    fork();
    fork();
    fork();
    printf("Done!\n");
  }

  **Five lines**
Alternate first question

How many lines will be printed by the following program?

#include <stdio.h>
main() {
    fork();
    printf("Start\n");
    fork();
    fork();
    printf("Done!\n");
}
Answer

- How many lines will be printed by the following program?
- `#include <stdio.h>
  main() {
    fork();
    printf("Start\n");
    fork();
    fork();
    printf("Done!\n");
  }

  *Six lines*
Second question

- Where does UNIX store the *file names*?
Answer

- Where does UNIX store *file names*?
  - *In the directory entry/entries pointing to the i-node of each individual file.*
Third question

- What are *UNIX soft links*?
Answer

■ What are **UNIX soft links**?

- **UNIX soft links** are special entities within the file system that point to other files, much like Windows shortcuts.

- They are also called **symbolic links**.
- They can cross partition boundaries.
Fourth question

- Where does UNIX store access control lists?
Answer

- Where does UNIX store access control lists?
  - *In the i-node of each file*
Fifth Question

- Why does the Fast File System subdivide each disk partition into \textit{cylinder groups}?
Answer

- Why does the Fast File System subdivide each disk partition into **cylinder groups**?

  - *It minimizes disk arm motions because each cylinder group has its own fragment of the i-node table. As a result most file blocks reside closer to their file i-node.*
Sixth question

Why does FFS use blocking writes to implement all metadata updates?
Answer

- Why does FFS use blocking writes to implement all metadata updates?

  - To guarantee the consistency of the file system
  - And the durability of metadata updates
Seventh question

While the page replacement policies of VMS and Mach are fairly similar, each of them has a strong point that the other doesn't have.

- What is the main advantage of the VMS policy compared to the Mach policy?

- What is the main advantage of the Mach policy compared to the VMS policy?
Answer

- While the page replacement policies of VMS and Mach are fairly similar, each of them has a strong point that the other doesn't have.

  - What is the main advantage of the VMS policy compared to the Mach policy?
    - **VMS supports real-time processes**

  - What is the main advantage of the Mach policy compared to the VMS policy?
Answer

While the page replacement policies of VMS and Mach are fairly similar, each of them has a strong point that the other doesn't have.

- What is the main advantage of the VMS policy compared to the Mach policy?
  - **VMS supports real-time processes**

- What is the main advantage of the Mach policy compared to the VMS policy?
  - **Mach policy is easier to tune**
Eighth question

- What is the main advantage of *mapped files*?

- Which *memory object* is associated with each mapped file?

- What should be the *inheritance* attribute of a mapped file?
Answer

- What is the main advantage of *mapped files*?
  - *They eliminate context switches by bringing file blocks directly into the address space of the process accessing them*
Answer

- Which *memory object* is associated with each mapped file?

  - The file being accessed
Answer

- What should be the *inheritance* attribute of a mapped file?

  - Shared