

۲.۱۲) سرویس‌ها و توابع تهیه‌شده به وسیله سیستم‌عامل به دو دسته اصلی می‌توانند تقسیم شوند. به طور خلاصه این دو دسته را توصیف کرده و در مورد اختلاف آنها بحث کنید.

الف) دسته‌ی اول این سرویس‌ها جهت اجرا و تاکید روی محافظت از فرآیندهای مختلف است که هم‌زمان روی سیستم اجرا می‌شوند. فرآیندها فقط باید به مکان‌هایی از حافظه که برای خودشان است دسترسی داشته باشند و همچنین نباید بتوانند فایل‌هایی که برای سایر کاربران است را خراب کنند. فرآیندها به طور مستقیم (بدون دخالت سیستم‌عامل) نباید به سخت‌افزار و دستگاه‌ها دسترسی داشته باشند.

ب) دسته‌ی دوم از خدمات تهیه‌شده توسط یک سیستم‌عامل، ارائه‌ی امکانات جدیدی است که به طور مستقیم توسط سخت‌افزار مورد نیاز پشتیبانی نمی‌شود. مانند حافظه‌ی مجازی<sup>۱</sup> و سیستم‌های فایل.

۲.۱۵) پنج فعالیت عمده از یک سیستم‌عامل در مورد مدیریت فایل چه هستند؟

۱- ساختن و پاک کردن فایل‌ها

۲- ساختن و پاک کردن پوشه‌ها

۳- پشتیبانی از عملیات اولیه برای دست‌کاری فایل‌ها و پوشه‌ها

۴- تطبیق<sup>۲</sup> فایل‌ها روی حافظه‌ی ثانویه

۵- تهیه‌ی نسخه‌ی پشتیبان<sup>۳</sup> از فایل‌ها روی حافظه‌های پایدار و غیرشکننده<sup>۴</sup>

۲.۱۷) آیا این امکان وجود دارد که کاربر یک مفسر فرمان جدید با استفاده از واسط فراخوانی سیستم تهیه شده به وسیله سیستم‌عامل تولید نماید؟

کاربر می‌تواند با استفاده از واسط فراخوانی-سیستمی که توسط سیستم‌عامل ارائه می‌شود، یک مفسر فرمان جدید را توسعه دهد. مفسر فرمان به کاربر اجازه می‌دهد تا فرآیندها را ایجاد و یا مدیریت کند، همچنین این امکان را فراهم می‌کند که راه ارتباطی کاربر و فرآیند را انتخاب کند. از آنجایی که تمام این قابلیت و عملکردها توسط فراخوانی‌های سیستم با یک برنامه در سطح کاربر قابل دسترسی هستند، امکان تولید و توسعه‌ی یک مفسر فرمان جدید برای کاربر وجود دارد.

۲.۲۱) مزیت اصلی روش ریزهسته در طراحی سیستم‌عامل چیست؟ در یک معماری ریزهسته برنامه‌های کاربر و سرویس‌ها چگونه با هم تعامل دارند؟ عیب استفاده از ریزهسته چیست؟  
مزیت‌ها:

الف) اضافه کردن سرویس جدید نیازی به تغییر و اصلاح هسته ندارد.

ب) چون بیشتر برنامه‌ها در حالت کاربر اجرا می‌شوند، امنیت بالاتری دارد.

ج) معمولاً یک طراحی و عمل‌کرد ساده‌تر منجر به ساخت سیستم‌عامل مطمئن‌تری می‌شود.

<sup>۱</sup> Virtual Memory

<sup>۲</sup> Map

<sup>۳</sup> Backup

<sup>۴</sup> Non-volatile

در معماری ریزهسته، برنامه‌های کاربر و سرویس‌ها برای تعامل از یک مکانیزم ارتباط بین-فرآیندی مانند پیام‌رسانی استفاده می‌کنند. این پیام‌ها توسط سیستم‌عامل منتقل می‌شود.  
عیب‌ها:

عیب اصلی استفاده از این روش سرریزهای مربوط به ارتباط بین پردازش‌ها و استفاده زیاد و مکرر از توابع پیام‌رسان سیستم‌عامل برای تعامل فرآیند کاربر و سرویس سیستم با یکدیگر است.

**پژوهش:** درباره‌ی سیستم‌عامل‌های MINIX 3 و ReactOS تحقیق کنید و ویژگی‌های آن‌ها مانند هدف از تولید، ساختار هسته سیستم‌عامل، پردازنده‌های پشتیبانی‌شده و حجم فعلی کد هسته را توضیح دهید.

### MINIX 3



MINIX 3 یک سیستم‌عامل جدید منبع-باز است که بسیار قابل‌اعتماد، انعطاف‌پذیر و ایمن طراحی شده است. این سیستم‌عامل مبتنی بر نسخه‌های قبلی MINIX است (MINIX‌های 1 و 2 به عنوان ابزار تدریس در نظر گرفته شدند)، اما از نظر اساسی از بسیاری جهات متفاوت است.

هدف MINIX 3 این است که در کامپیوترهایی که منابع محدود دارند و یا کامپیوترهای جاسازی‌شده برای کاربردهایی که نیاز به قابلیت اطمینان بالایی دارند، بتواند به عنوان یک سیستم‌عامل جدی استفاده شود.

این سیستم‌عامل جدید بسیار کوچک (از نظر حجمی) است، با بخشی که در حالت هسته با کم‌تر 4000 خط کد اجرا می‌شود. قسمت‌هایی که در حالت کاربر اجرا می‌شوند به ماژول‌های کوچک‌تری تقسیم می‌شوند که به خوبی از یکدیگر عایق‌بندی و جدا شده‌اند. به عنوان مثال، هر درایور دستگاه به عنوان یک فرآیند در حالت کاربر جداگانه اجرا می‌شود، بنابراین اشکالی در درایور (که تا الان بزرگ‌ترین منبع اشکالات در هر سیستم‌عامل بوده)، نمی‌تواند کل سیستم‌عامل را خراب کند. این سیستم‌عامل روی پردازنده‌های x86 و ARM اجرا می‌شود، با NetBSD سازگار است و هزاران پکیج NetBSD را اجرا می‌کند.

\* در آخرین کامیت MINIX 3 در سایت openhub این پروژه شامل 7,477,633 خط کد، 2,169,916 خط کامنت و 1,374,128 خط خالی بوده است که 49% آن‌ها با زبان C، 34% با ++C و 17% با سایر زبان‌ها نوشته شده است.

### ReactOS



پروژه‌ی ReactOS یک پیاده‌سازی یک سیستم‌عامل سازگار با ویندوز XP / 2003 است. ReactOS با استفاده از یک معماری مشابه و ارائه‌ی یک رابط عمومی کامل و معادل آن، دستیابی به سازگاری کامل باینری با برنامه‌های کاربردی و درایورهای دستگاه را برای سیستم‌عامل‌های NT و XP را ممکن می‌کند. این سیستم‌عامل نیز بسیار کم‌حجم (500MB HDD و 96MB RAM) است.

\* \* در آخرین کامیت ReactOS در سایت openhub این پروژه شامل 11,955,213 خط کد، 1,959,003 خط کامنت و 2,179,669 خط خالی بوده است که 90% آن با زبان C، 9% آن با ++C، و 1% آن با زبان‌های برنامه‌نویسی دیگر تهیه شده است.