بِسۡمِ اللّٰہِ الرَّحۡمٰنِ الرَّحِیۡمِ

اللہ کے نام سے شروع جو بڑا مہربان نہایت رحم فرمانے والا ہے۔

کمپیوٹر سائنس اینڈ انٹرپرینیورشپ

10

نیشنل کریکولم آف پاکستان 2023ء (نظرِ ثانی شدہ) کے مطابق

Web version of PECTAA Textbook
Not for Sale

پنجاب ایجوکیشن، کریکولم، ٹریننگ اینڈ اسسمنٹ اتھارٹی

PECTAA
Transformation, Innovation and Excellence

مصنّفین

- پروفیسر ڈاکٹر محمد عاطف چٹھہ
ڈین فیکلٹی آف کمپیوٹر سائنس
لاہور گیریژن یونیورسٹی لاہور
- پروفیسر ڈاکٹر ارشد علی
فاسٹ سکول آف کمپیوٹنگ، یونیورسٹی آف کمپیوٹر
اینڈ امرجنگ سائنسز (NUCES) لاہور کیمپس
- ڈاکٹر عبد الستار
ڈیپارٹمنٹ آف کمپیوٹر سائنس
لاہور گیریژن یونیورسٹی لاہور
- پروفیسر ڈاکٹر اشفاق احمد
لاہور گیریژن یونیورسٹی، لاہور
- پروفیسر ثاقب محمود
لاہور گیریژن یونیورسٹی، لاہور
- ذوالفقار علی کلاچی
ہیڈ ماسٹر، گورنمنٹ ہائی سکول، کوٹ ہیبت، ڈی جی خان

ایڈیٹرز

- پروفیسر ڈاکٹر مدثر نصیر
ڈیپارٹمنٹ آف کمپیوٹر سائنس اینڈ آئی ٹی، یونیورسٹی آف لاہور، لاہور
- پروفیسر ثاقب عبید
انسٹیٹیوٹ آف کمپیوٹر سائنس، خواجہ فرید یونیورسٹی آف انجینئرنگ اینڈ آئی ٹی، رحیم یار خان

مترجم

- کاشف شہزاد چودھری گورنمنٹ مسلم ہائی سکول نمبر 2، سول لائنز، لاہور

ریویو کمیٹی

- پروفیسر محمد فہیم
گورنمنٹ گریجویٹ کالج، گلبرگ لاہور
- پروفیسر مہوش پونم
گورنمنٹ ایسوسی ایٹ کالج برائے خواتین، گوالمنڈی لاہور
- انجم حسین
اسسٹنٹ ایجوکیشن آفیسر، مرکز خان پور، تحصیل کوٹ چھٹہ
- پروفیسر محمود احمد چودھری
دی گریجویٹ کالج، شادمان، لاہور

ریویو کمیٹی برائے اُردو میڈیم

- اعجاز حسین SST(CS/IT)
گورنمنٹ بوائز ہائی سکول، گوالمنڈی لاہور
- صائمہ SST(CS/IT)
گورنمنٹ ماڈل گرلز ہائی سکول، سمن آباد لاہور
- محمد نوید الرحمٰن SST(CS/IT)
گورنمنٹ ہائی سکول چوہنگ، ملتان روڈ، لاہور
- سید مدثر حسین SST(CS/IT)
گورنمنٹ اسلامیہ ہائی سکول، سنت نگر، لاہور

نگرانِ طباعت

- جہاں زیب خان، اسسٹنٹ ڈائریکٹر، پنجاب ایجوکیشن، کریکولم، ٹریننگ اینڈ اسسمنٹ اتھارٹی، لاہور

ڈائریکٹر (C&C) — عامر ریاض
ڈپٹی ڈائریکٹر (C-S) — سیّد صغیر الحسنین ترمذی
انچارج آرٹ سیل — عائشہ صادق
لے آؤٹ اینڈ ڈیزائننگ — طاہر سرفراز بٹ
السٹریٹنز — آیت اللہ، طاہر سرفراز بٹ

تجرباتی ایڈیشن

ناشر: پنجاب ایجوکیشن کریکولم ٹریننگ اینڈ اسیسمنٹ اتھارٹی، لاہور۔
مطبع: سلور کلاسک پرنٹنگ اینڈ پیکیجنگ، لاہور

DATE OF PRINTING	PMIU DANISH SCHOOL	PEF	MLWC	PWWF	TOTAL QUANTITY
Feb, 2026	187,921	15,285	127	94	203,427

Web version of PECTAA Textbook
Not for Sale

یہ کتاب پاکستان کے نظرثانی شدہ قومی نصاب 2023ء پر مبنی ہے۔ جملہ حقوق بحق پنجاب ایجوکیشن، کریکولم، ٹریننگ اینڈ اسسمنٹ اتھارٹی محفوظ ہیں۔اس کتاب کا کوئی حصہ نقل یا ترجمہ نہیں کیا جاسکتا نہ ہی ٹیسٹ پیپر، گائیڈ بکس، خلاصہ جات، نوٹس یا امدادی کتب کی تیاری میں استعمال کیا جاسکتا ہے۔

Web version of PECTAA Textbook
Not for Sale

فہرست

یونٹ	عنوانات	صفحہ نمبر
1	آپریٹنگ سسٹمز ساخت اور خدمات	1
2	سسٹم ریکوری اینڈ مینٹیننس	19
3	تعارف برائے پائتھن پروگرامنگ	40
4	پائتھن میں کنٹرول اسٹرکچر	57
5	ڈیٹا سائنس کا تعارف	73
6	مصنوعی ذہانت (AI) اور مشین لرننگ (ML) کا تعارف	95
7	مصنوعی ذہانت (AI) کے اطلاقات	109
8	ڈیجیٹل انٹرپرینیورشپ	125
	سوالیہ پرچے کی تیاری بارے ہدایات	
	ماڈل سوالیہ پرچہ	

یونٹ

1

آپریٹنگ سسٹمز ساخت اور خدمات

حاصلاتِ تعلم: اس باب کے اختتام تک، آپ اس قابل ہو جائیں گے کہ:

- آپریٹنگ سسٹم کی تعریف کر سکیں اور اس کے کردار کی وضاحت کر سکیں کہ یہ صارف اور ہارڈویئر کے درمیان بطور رابطہ (Interface) کیسے کام کرتا ہے۔
- متعدد صارف اکاؤنٹس کو سنبھالنے میں آپریٹنگ سسٹم کی ذمہ داریوں کی شناخت کر سکیں۔
- کرنل (Kernel) اور شیل (Shell) کے درمیان فرق بتا سکیں، بشمول ان کے مخصوص افعال۔
- آپریٹنگ سسٹم کی تہوں (OS Layers) اور ہر تہہ ایک دوسرے کے ساتھ کیسے باہمی رابطہ کرتی ہے، کی وضاحت کر سکیں۔
- آپریٹنگ سسٹم کے کردار میں سسٹم لائبریریوں اور ڈیوائس ڈرائیورز کے کردار کی شناخت کر سکیں۔
- پروسیس لائف سائیکل کے مراحل (تخلیق، عملدرآمد، اختتام) کی مثالوں کے ساتھ وضاحت کر سکیں اور فرسٹ-کم، فرسٹ-سروڈ (FCFS) پروسیس شیڈولنگ کو لاگو کر سکیں۔
- ملٹی ٹاسکنگ اور ہم آہنگی (Concurrency) کی وضاحت کر سکیں، اور ایک حقیقی دنیا کی مثال دے سکیں۔
- پرائمری میموری (RAM) اور ورچوئل میموری کے درمیان فرق کر سکیں۔
- پروسیس اور تھریڈ کی تعریف کر سکیں، اور وضاحت کریں کہ تھریڈز ایک پروسیس کے اندر وسائل کو کیسے شیئر کرتے ہیں۔
- مثالیں دیں کہ کس طرح ملٹی تھریڈنگ (Multithreading) ایپلیکیشنز میں کارکردگی کو بہتر بناتی ہے۔
- سسٹم کال (System Call) کی تعریف کر سکیں اور اس کا مقصد اور اقسام کی وضاحت کر سکیں۔
- فائل سسٹم، فائلیں، فولڈرز، اور میٹاڈیٹا کی تعریف کر سکیں، اور ڈیٹا کو منظم کرنے میں ان کے کردار کی وضاحت کر سکیں۔
- یہ اجزاء مل کر کس طرح صارف کی درخواست کو ڈیوائس سے سرور تک اور پھر واپس پراسیس کرتے ہیں، کی وضاحت کر سکیں۔

تعارف

اس باب کے اختتام تک، آپ سمجھ جائیں گے کہ ایک آپریٹنگ سسٹم کس طرح درپردہ کام کرتا ہے تاکہ وسائل کا انتظام کرے، کارکردگی کو برقرار رکھے، ڈیٹا کو محفوظ رکھے، اور موثر اور کارآمد سسٹم کی کارکردگی کو یقینی بنانے کے لیے ضروری پس منظر کے افعال انجام دے۔ اس باب میں ہم دیکھیں گے:

- آپریٹنگ سسٹم کا آرکیٹیکچر اور کمپیوٹر کے مرکزی کنٹرولر کے طور پر اس کا کردار۔ پراسس مینجمنٹ کا تصور جیسا کہ پراسس لائف سائیکل، ملٹی ٹاسکنگ اور شیڈولنگ۔
- میموری مینجمنٹ، ملٹی تھریڈنگ، اور ہارڈویئر وسائل تک محفوظ طریقے سے رسائی کے لیے سسٹم کالز کا استعمال۔
- فائل سسٹم کی تنظیم اور آپریٹنگ سسٹم کی اہم اقسام، بشمول ریئل ٹائم، ایمبیڈڈ، نیٹ ورک، اور موبائل آپریٹنگ سسٹم۔

Web version of PECTAA Textbook
Not for Sale

1.1 آپریٹنگ سسٹم (OS) کا تعارف

آپریٹنگ سسٹم (OS) ایک قسم کا سسٹم سافٹ ویئر ہے جو کمپیوٹر کے ہارڈ ویئر کو منظم کرتا ہے، ایپلیکیشن سافٹ ویئر کو چلاتا ہے، اور صارف (User) کو انٹرفیس فراہم کرتا ہے۔ اس کی مثالیں Windows, macOS, Linux, Android اور iOS ہیں، جن کے اپنے بنیادی کام ہوتے ہیں۔ یہ باب اس بات پر توجہ دیتا ہے کہ آپریٹنگ سسٹم کس طرح کمپیوٹر سسٹم کے مرکزی کنٹرولر کے طور پر کام کرتا ہے اور صارف اور ہارڈ ویئر کے درمیان ہموار اور محفوظ رابطہ قائم رکھتا ہے، خاص طور پر جب ایک سے زیادہ صارفین (multi-user) سسٹم استعمال کر رہے ہوں۔

آپریٹنگ سسٹم بطور مرکزی سسٹم کنٹرولر (OS as the Central System Controller)

ایک آپریٹنگ سسٹم کمپیوٹر کے لیے ٹریفک کنٹرولر کی طرح کام کرتا ہے۔ یہ فیصلہ کرتا ہے کہ کون سا کام پہلے کیا جانا چاہیے، کمپیوٹر کی میموری کس طرح استعمال ہوتی ہے، اور ایک مخصوص وقت میں کون سی ڈیوائسز (جیسے پرنٹر یا اسپیکر) فعال ہونی چاہئیں۔ مختصراً، ہم کہہ سکتے ہیں کہ OS اس بات کو یقینی بناتا ہے کہ کمپیوٹر آسانی سے کام کرتا ہے، یہاں تک کہ جب ایک ہی وقت میں متعدد کام چل رہے ہوں۔

صارف اور ہارڈ ویئر کے درمیان باہمی رابطہ کا کردار (Role in User-Hardware Interaction)

کمپیوٹر ہارڈ ویئر براہ راست انسانی زبان کو نہیں سمجھ سکتا۔ آپریٹنگ سسٹم صارف اور ہارڈ ویئر سسٹم کے درمیان ایک مترجم کے طور پر کام کرتا ہے۔
مثال: جب آپ کسی آئیکن پر کلک کرتے ہیں یا کچھ ٹائپ کرتے ہیں، تو OS ان افعال کو ان ہدایات میں بدل دیتا ہے جنہیں ہارڈ ویئر سمجھ سکتا ہے۔ یہ لوگوں کو ہارڈ ویئر کے کام کرنے کا طریقہ سیکھے بغیر کمپیوٹر کو آسانی سے استعمال کرنے کی اجازت دیتا ہے۔

کیا آپ جانتے ہیں؟

انسان اردو یا انگریزی جیسی زبانیں بولتے ہیں، جبکہ کمپیوٹر صرف بائنری زبان یا آن اور آف سوئچ یعنی 0 اور 1 کو سمجھتے ہیں۔
مثال: جب آپ اپنے کی بورڈ پر حرف 'A' دباتے ہیں، تو آپریٹنگ سسٹم اسے ایک خاص بائنری کوڈ میں تبدیل کرتا ہے:
بائنری میں حرف 'A'، 01000001_2 ہے۔
یہ کوڈ پھر ہارڈ ویئر کو بھیجا جاتا ہے، جو اسکرین پر 'A' کو ظاہر کرتا ہے۔ یہ عمل اتنی تیزی سے ہوتا ہے کہ صارف کو فوری دکھائی دیتا ہے۔

ملٹی یوزر ماحول میں ذمہ داریاں

اسکولوں، دفاتر، یا آن لائن سسٹمز جیسی جگہوں پر، بہت سے لوگ ایک ہی کمپیوٹر سسٹم استعمال کر سکتے ہیں۔ آپریٹنگ سسٹم اس بات کو یقینی بناتا ہے کہ:

- الگ الگ یوزر اکاؤنٹس بناتا ہے اور ان کا انتظام کرتا ہے۔
- ہر صارف کی فائلوں اور معلومات (Files and Information) کو محفوظ اور نجی (Private) رکھتا ہے۔

Web version of PECTAA Textbook Not for Sale

- کمپیوٹر کے وسائل کو منصفانہ طور پر تقسیم کرتا ہے تاکہ کوئی ایک صارف سسٹم کی کارکردگی (System Functionality) کو متاثر نہ کرے۔
- مشترکہ وسائل کو غیر مجاز رسائی سے محفوظ رکھا جائے۔

یوزر اکاؤنٹس بنانا اور انہیں منظم کرنا (Creating and Managing User Accounts)

آپریٹنگ سسٹم مختلف صارفین کے لیے الگ اکاؤنٹس کی اجازت دیتا ہے، ہر ایک یوزر کے اپنے ڈیسک ٹاپ، فائلوں، سیٹنگز، اور پاس ورڈز کے ساتھ۔

مثال: کمپیوٹر لیب میں، طلباء اسکول کی طرف سے فراہم کردہ یوزر نیم اور پاس ورڈ کے ساتھ لاگ ان ہوتے ہیں۔ یہ ہر طالب علم کے کام کو الگ اور نجی رکھتا ہے۔ ونڈوز جیسے زیادہ تر جدید OS میں، Settings یا Control Panel کے ذریعے ایک نیا اکاؤنٹ بنایا جاسکتا ہے، جہاں User Accounts کو منتخب کر کے، Add New User کا انتخاب کیا جاتا ہے، اور یوزر نیم، پاس ورڈ، اور اکاؤنٹ کی قسم (Standard, Administrator, Guest) جیسی تفصیلات درج کی جاتی ہیں۔

1.2 آپریٹنگ سسٹم کا آرکیٹیکچر

ایک آپریٹنگ سسٹم کا آرکیٹیکچر وہ طریقہ ہے جس سے اس کے حصے منظم ہوتے ہیں اور وہ ایک ساتھ کام کرتے ہیں۔ ہر حصے کا ایک خاص کردار ہوتا ہے، اور وہ تمام مل کر کمپیوٹر کو آسانی سے کام کرنے کے قابل بناتے ہیں۔ یہ بالکل ایک اسکول کی طرح ہے جس میں مختلف شعبے مخصوص فرائض انجام دیتے ہیں لیکن ادارے کے ہموار کام کے لیے مل کر کام کرتے ہیں۔

کرنل بمقابلہ شیل (Kernel vs. Shell)

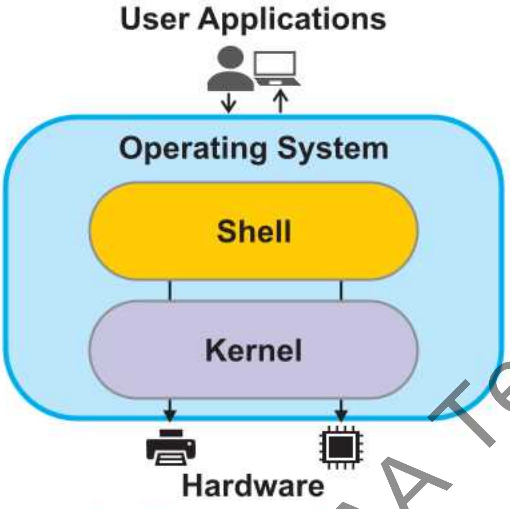
کرنل (Kernel): کرنل آپریٹنگ سسٹم کا بنیادی حصہ ہے جو براہ راست کمپیوٹر کے ہارڈ ویئر اور سسٹم کے وسائل، جیسے CPU، میموری، اور ڈیوائسز کو کنٹرول کرتا ہے جیسا کہ شکل 1.1 میں دکھایا گیا ہے۔ یہ فیصلہ کرتا ہے کہ مختلف پروگرام ان وسائل کو کب اور کیسے استعمال کر سکتے ہیں۔

مثال: جب آپ کوئی فائل کھولتے ہیں، تو کرنل اسے ہارڈ ڈرائیو سے پڑھنے اور اسکرین پر بھیجنے کے عمل کا انتظام کرتا ہے۔
کرنل کا کام کار کے انجن جیسا ہے، جبکہ شیل، ڈیش بورڈ جیسا ہے اور اسیسریز اسٹیئرنگ وہیل جیسی ہیں۔

شیل (Shell): شیل OS کا بیرونی حصہ ہے جو صارف کے ساتھ تعامل کرتا ہے۔ یہ صارف سے کمانڈز وصول کرتا ہے اور انہیں کرنل تک پہنچاتا ہے۔ شیلز کی دو اقسام ہیں:

- گرافیکل شیلز (Graphical shells): (جیسے ونڈوز ڈیسک ٹاپ) جہاں آپ آئیکنز پر کلک کرتے ہیں اور مینیوز کا استعمال کرتے ہیں۔
- کمانڈ-لائن شیلز (Command-line shells): (جیسے کمانڈ پرامپٹ یا ٹرمینل) جہاں آپ ہدایات ٹائپ کرتے ہیں۔

Web version of PECTAA Textbook Not for Sale



شکل 1.1: آپریٹنگ سسٹم میں صارف، شیل، اور کرنل کے درمیان تعلق کو واضح کرنے والی ڈایاگرام۔

OSI کی لیئرز اور ماڈیولر ڈیزائن

آپریٹنگ سسٹمز میں، ڈیزائن کو لیئرز میں تقسیم کیا جاتا ہے، جہاں ہر لیئر کا ایک مخصوص کام ہوتا ہے:

- **لور لیئرز (Lower layers):** براہ راست ہارڈ ویئر ڈیوائسز، جیسے CPU، میموری، اور اسٹوریج اور ہارڈ ڈرائیو کے ساتھ کام کرتی ہیں۔
- **مڈل لیئرز (Middle layers):** وسائل کا انتظام کرتی ہیں اور یقینی بناتی ہیں کہ ضرورت پڑنے پر پروگرام انہیں استعمال کر سکیں۔
- **اَپر لیئرز (Upper layers):** ایپلی کیشنز کو چلاتی ہیں اور وہ انٹرفیس فراہم کرتی ہیں جسے صارف اسکرین پر دیکھتا ہے۔
- ہر لیئر اپنے نیچے والی لیئر پر منحصر ہوتی ہے۔ یہ ڈیزائن آپریٹنگ سسٹم کے پورے سسٹم کو تبدیل کیے بغیر انتظام، مرمت، اور بہتر بنانے میں آسانی فراہم کرتا ہے۔

مثال: آپریٹنگ سسٹم کے لیئرڈ آرکیٹیکچر کو سمجھنے کے لیے ایک اسکول سسٹم کی مثال استعمال کی جاسکتی ہے:

- سپورٹ اسٹاف (Support Staff): جیسے چوکیدار اور صفائی کرنے والے نچلی سطح پر کام کرتے ہیں، تاکہ اسکول تیار رہے (جیسے لوئیر ہارڈ ویئر کے ساتھ کام کر رہی ہیں)۔
- انتظامیہ (Administration): وسائل، شیڈول، اور قوانین کا انتظام کرتی ہے (جیسے مڈل لیئر میموری اور اسٹوریج کا انتظام کر رہی ہیں)۔

اساتذہ اور طلباء (Teachers and Students): ان انتظامات کو پڑھانے اور سیکھنے کے لیے استعمال کرتے ہیں (جیسے اَپر لیئرز ایپلی کیشنز کو
- چلاتی ہیں اور صارفین کے ساتھ تعامل کرتی ہیں)۔

NOT FOR SALE-PESRP
Web version of PECTAA Textbook Not for Sale

سسٹم لائبریریز اور ڈیوائس ڈرائیورز (System Libraries and Device Drivers)

سسٹم کی لائبریریاں تیار شدہ ہدایات کا مجموعہ ہیں جنہیں پروگرام عام کاموں کو انجام دینے کے لیے استعمال کر سکتے ہیں، جیسے فائلیں کھولنا یا اسکرین پر ٹیکسٹ دکھانا۔

مثال: جب کسی فوٹو ایڈیٹنگ ایپ کو کوئی تصویر کھولنے کی ضرورت ہوتی ہے، تو وہ فائل کو پڑھنے کے لیے آپریٹنگ سسٹم کی لائبریری کا استعمال کرتی ہے۔ اس طرح، ایپ کو تصاویر کھولنے کے لیے اپنا طریقہ اپنانے کی ضرورت نہیں پڑتی۔

ڈیوائس ڈرائیورز (Device Drivers) خاص پروگرام ہیں جو آپریٹنگ سسٹم کو ہارڈویئر ڈیوائسز، جیسے پرنٹرز، کی بورڈز، اور گرافکس کارڈز کے ساتھ بات چیت کرنے کی اجازت دیتے ہیں۔

سرگرمی:

مقصد: اپنے کمپیوٹر کے ذریعے استعمال ہونے والے شیل، ڈیوائس ڈرائیور، اور سسٹم لائبریری کی شناخت کریں۔

طریقہ کار:

1. اپنے کمپیوٹر پر شیل کا پتہ لگائیں (ڈیسک ٹاپ یا کمانڈ-لائن)۔
2. Device Manager (ونڈوز) System Information (میک / لینکس) کھولیں۔
3. ایک ہارڈویئر ڈیوائس (مثلاً، پرنٹر، کی بورڈ) تلاش کریں اور اس کے ڈرائیور کا نام نوٹ کریں۔
4. ایک سادہ پروگرام (مثلاً، کیلکولیٹر) کھولیں اور ایک ایسا کام پہچانیں جو یہ انجام دیتا ہے جس میں سسٹم لائبریری کا استعمال ہو سکتا ہے (مثلاً، نمبرز کو ظاہر کرنا)۔
5. اپنی معلومات کو ان کالموں کے ساتھ ایک ٹیبل میں ریکارڈ کریں:
   آپریٹنگ سسٹم کا کمپونٹ — (OS Component) جیسے: شیل، ڈیوائس ڈرائیور، یا سسٹم لائبریری
   آپ کی مثال — (Your Example) مثلاً: ونڈوز ڈیسک ٹاپ، ایچ پی پرنٹر ڈرائیور، کیلکولیٹر ایپ
   مقصد / کردار — (Purpose / Role) یہ نظام میں کیا کردار ادا کرتا ہے یا کیا کام انجام دیتا ہے

1.3 آپریٹنگ سسٹم میں پروسیس مینجمنٹ

آپریٹنگ سسٹم کمپیوٹر پر چلنے والے تمام پروگراموں کو منظم کرنے کا ذمہ دار ہے۔ اس تناظر میں، ان چلنے والے پروگراموں کو پروسیسز (Processes) کہا جاتا ہے۔ پروسیس مینجمنٹ اس بات کو یقینی بناتا ہے کہ ہر پروسیس کو وہ وسائل ملیں جن کی اسے ضرورت ہے، یہاں تک کہ جب متعدد پروسیسز فعال ہوں۔ پروسیس مینجمنٹ ایک آپریٹنگ سسٹم کے سب سے اہم کاموں میں سے ایک ہے کیونکہ یہ یقینی بناتا ہے کہ تمام پروسیسز ایک دوسرے کو متاثر کیے بغیر درست کام کریں۔

پروسیس لائف سائیکل

ایک پروسیس اپنی لائف سائیکل کے دوران کئی مراحل سے گزرتا ہے:

1. تخلیق (Creation): یہ تب ہوتا ہے جب آپ کوئی بھی پروگرام شروع کرتے ہیں (جیسے MS Word)۔

Web version of PECTAA Textbook
Not for Sale

OS پروگرام کو میموری میں لوڈ کرتا ہے اور اسے مطلوبہ وسائل (جیسے CPU کا وقت اور میموری) دیتا ہے۔

2. عمل در آمد (Execution): پروسیس فعال طور پر چل رہا ہے اور مؤثر طریقے سے کام انجام دے رہا ہے۔

3. اختتام (Termination): پروسیس اپنا کام مکمل کرتا ہے اور صارف یا سسٹم کے ذریعہ بند کر دیا جاتا ہے۔ OS وسائل کو آزاد کر دیتا ہے تاکہ انہیں دوسرے پروسیسز استعمال کر سکیں۔

مثال: آپ کے موبائل یا کمپیوٹر پر ویب براؤزر کھولنے کا پروسیس لائف سائیکل (جیسے Google Chrome):

- **تخلیق:** جب صارف براؤزر آئیکن پر کلک کرتا ہے تو پراسس شروع ہوتا ہے۔ آپریٹنگ سسٹم پروگرام کو میموری میں لوڈ کرتا ہے اور مطلوبہ وسائل مختص کرتا ہے۔
- **عملدرآمد:** براؤزر کام انجام دیتا ہے، جیسے ویب پیجز لوڈ کرنا، میڈیا دکھانا، اور صارف کے اعمال کا جواب دینا۔
- **اختتام:** جب براؤزر بند ہو جاتا ہے تو اختتام واقع ہوتا ہے۔ آپریٹنگ سسٹم پروسیس کو روکتا ہے اور اس کے وسائل کو دیگر پراسسز کے استعمال کے لیے فارغ کرتا ہے۔

ملٹی ٹاسکنگ اور ہم آہنگی (Multitasking and Concurrence)

جدید آپریٹنگ سسٹمز بہت سے پروسیسز کو اتنے مؤثر طریقے سے منظم کرتے ہیں کہ ایسا لگتا ہے کہ وہ سب ایک ہی وقت میں چل رہے ہیں۔

- ملٹی ٹاسکنگ (Multitasking): آپریٹنگ سسٹم ایک ہی صارف کے ذریعہ ایک ہی وقت میں ایک سے زیادہ پروگراموں کو کھولنے اور استعمال کرنے کی اجازت دیتا ہے۔ آپ ضرورت پڑنے پر آسانی سے ان کے درمیان سوئچ کر سکتے ہیں۔

مثال: آپ موسیقی سن سکتے ہیں، ایک دستاویز کو کھلا رکھ سکتے ہیں، اور انٹرنیٹ براؤز کر سکتے ہیں، ضرورت کے مطابق ان کے درمیان سوئچ کر سکتے ہیں۔

- ہم آہنگی (Concurrency): ایک آپریٹنگ سسٹم میں ایک سے زیادہ پروسیسز ایک ہی وقت میں فعال ہوتے ہیں، لیکن CPU انہیں انتہائی تیزی سے دائروی عمل میں ایک کے بعد ایک پروسیس کرتا ہے۔

مثال: جیسے ایک شیف تین پکوان تیار کر رہا ہو، ایک پر تھوڑے وقت کے لیے کام کر رہا ہو، پھر دوسرے پر منتقل ہو رہا ہو، اور اس عمل کو دہراتا ہو، CPU پروسیسز کے درمیان اتنی تیزی سے سوئچ کرتا ہے کہ صارف کو کوئی تاخیر محسوس نہیں ہوتی۔

پروسیس شیڈولنگ کے تصورات (Process Scheduling Concepts)

ہم نے سیکھا کہ بہت سے پروسیسز ایک ہی وقت کی مدت کے دوران چل سکتے ہیں، لیکن CPU ان پر باری باری بہت تیزی سے لیکن ایک وقت میں ایک پر کام کرتا ہے۔ چونکہ CPU ایک ہی وقت میں تمام پروسیسز کو نہیں چلا سکتا، اس لیے آپریٹنگ سسٹم کو فیصلہ کرنا چاہیے کہ:

- کون سا پروسیس پہلے چلنا چاہیے؟

NOT FOR SALE-PESRP
Web version of PECTAA Textbook Not for Sale

- ہر پروسیس کو کتنی دیر تک چلنا چاہیے؟

یہ فیصلہ سازی کا عمل شیڈولنگ (Scheduling) کہلاتا ہے۔ آپریٹنگ سسٹمز شیڈولنگ کے مختلف طریقے استعمال کرتے ہیں۔ لیکن اس باب میں، ہم صرف ایک سب سے آسان شیڈولنگ تکنیک، فرسٹ کم، فرسٹ سروڈ (First-Come, First-Served - FCFS) کو دیکھیں گے۔

FCFS طریقہ کار میں، CPU کاموں کو ان کی آمد کے عین مطابق ترتیب میں پروسیس کرتا ہے۔ سب سے پہلے آنے والا پروسیس سب سے پہلے مکمل کیا جاتا ہے، اور اگلا پروسیس صرف پچھلے کے ختم ہونے کے بعد ہی شروع ہوتا ہے۔

- مثال: یہ ایک دکان کے کاؤنٹر پر قطار کی طرح ہے، قطار میں موجود پہلا گاہک پہلے سرو کیا جاتا ہے، پھر اگلا، اور اسی طرح۔

عددی مثال (Numerical Example): آئیے یہ دیکھنے کے لیے کچھ سادہ ڈیٹا استعمال کریں کہ FCFS شیڈولنگ عملی طور پر کیسے کام کرتی ہے۔

پروسیس کی آئی-ڈی (Process ID)	آمد کا وقت (Arrival Time)	درکار وقت (Time Needed)
P1	0 سیکنڈ	5 سیکنڈ
P2	1 سیکنڈ	3 سیکنڈ
P3	2 سیکنڈ	2 سیکنڈ

مرحلہ وار وضاحت:

1. 0 سیکنڈ پر:
   - P1 سب سے پہلے پہنچتا ہے، لہذا یہ فوری طور پر چلنا شروع کر دیتا ہے۔
   - اسے ختم ہونے کے لیے 5 سیکنڈ درکار ہیں
   - یہ 0s سے 5s تک چلتا ہے۔
2. 1 سیکنڈ پر:
   - P2 پہنچتا ہے جبکہ P1 ابھی بھی چل رہا ہے، لہذا P2 انتظار کرتا ہے۔
3. 2 سیکنڈ پر:
   - P3 پہنچتا ہے، لیکن P1 ابھی بھی چل رہا ہے اور P2 پہلے ہی انتظار کر رہا ہے۔
   - FCFS میں، P3 کو بھی P1 اور P2 دونوں کے مکمل ہونے تک انتظار کرنا پڑے گا۔
4. 5 سیکنڈ پر:
   - P1 ختم ہو جاتا ہے۔
   - CPU اب P2 شروع کرتا ہے، جسے مکمل ہونے کے لیے 3 سیکنڈ درکار ہیں۔

Web version of PECTAA Textbook
Not for Sale

NOT FOR SALE-PESRP

- P2 5s سے 8s تک چلتا ہے۔

5. 8 سیکنڈ پر:

- P2 ختم ہو جاتا ہے۔
- CPU اب P3 شروع کرتا ہے، جسے 2 سیکنڈ درکار ہیں۔
- P3 شروع ہوتا ہے اور 8s سے 10s تک چلتا ہے۔

عملدرآمد کی ٹائم لائن:

[P1: 0–5] → [P2: 5–8] → [P3: 8–10]

0 – 5	5 – 8	8 – 10
P1	P2	P3

P1: 0 سیکنڈ سے 5 سیکنڈ تک (اختتام 5 سیکنڈ پر)

P2: 5 سیکنڈ سے 8 سیکنڈ تک (اختتام 8 سیکنڈ پر)

P3: 8 سیکنڈ سے 10 سیکنڈ تک (اختتام 10 سیکنڈ پر)

مشاہدہ (Observation): اگرچہ P3 کی مدت سب سے کم ہے (صرف 2 سیکنڈ)، اسے اب بھی P1 اور P2 کے ختم ہونے تک انتظار کرنا پڑتا ہے کیونکہ وہ پہلے پہنچے تھے۔ یہ FCFS کی اہم خصوصیات میں سے ایک ہے؛ مختصر کام تاخیر کا شکار ہو سکتے ہیں اگر وہ طویل کاموں کے بعد پہنچیں۔

FCFS کے فوائد اور نقصانات (Advantages and Disadvantages of FCFS):

اگرچہ FCFS ایک سادہ اور منصفانہ شیڈولنگ طریقہ ہے، اس کے فوائد اور نقصانات دونوں ہیں:

فوائد (Advantages): سمجھنے اور لاگو کرنے میں آسان، کیونکہ پروسیسز کو ان کی آمد کے عین مطابق ترتیب میں سرو کیا جاتا ہے۔ کوئی بھی بعد میں آنے والا پروسیس پہلے آنے والے پروسس کے چلنے میں رکاوٹ نہیں ڈالتا۔

نقصانات (Disadvantages): مختصر پروسیسز کو طویل پروسیسز کے پیچھے قطار میں کھڑے ہونے کی صورت میں ایک طویل وقت تک انتظار کرنا پڑ سکتا ہے (Convoy Effect)، جو سسٹم کی مجموعی کارکردگی کو متاثر کر سکتا ہے۔

کیا آپ جانتے ہیں؟

آپ کے کمپیوٹر کا CPU ایک سیکنڈ میں لاکھوں بار پروسیسز کے درمیان سوئچ کر سکتا ہے۔ یہ اتنی تیزی سے ہوتا ہے کہ آپ ایک ہی وقت میں موسیقی چلا سکتے ہیں، ویب براؤز کر سکتے ہیں، اور فائلیں ڈاؤن لوڈ کر سکتے ہیں۔

Web version of PECTAA Textbook
Not for Sale

سرگرمی: مقصد: فرسٹ-کم، فرسٹ-سروڈ (FCFS) طریقہ کار کا استعمال کرتے ہوئے پروسیسز کی ترتیب اور تکمیل کا وقت معلوم کریں۔

طریقہ کار:

1. درج ذیل پروسیسز پر غور کریں:

پروسیس (Process)	آمد کا وقت (Arrival Time)	درکار وقت (Time Needed)
P1	0s	4s
P2	1s	5s
P3	2s	3s

2. FCFS طریقہ کار کے مطابق پروسیسز کو اس ترتیب میں ترتیب دیں جس میں وہ چلیں گے۔
3. ہر پروسیس کے لیے شروع ہونے اور ختم ہونے کے وقت کا حساب لگائیں۔

سوالات کے جواب دیں

- تمام پروسیسز کے ختم ہونے میں کتنا کل وقت لگا؟
- کس پروسیس کو سب سے زیادہ انتظار کرنا پڑا؟

Web version of PECTAA Textbook Not for Sale

1.4 میموری (Memory)

میموری کمپیوٹر سسٹم کا ایک بنیادی جزو ہے، جو پروسیسنگ کے لیے درکار ڈیٹا اور ہدایات کو محفوظ یا ذخیرہ کرنے کے لیے استعمال ہوتی ہے۔ یہ یقینی بناتی ہے کہ CPU پروگرام پر عملدر آمد کے دوران معلومات تک تیزی سے رسائی حاصل کر سکے۔

پرائمری میموری (Primary Memory - RAM)

پرائمری میموری، جسے رینڈم ایکسیس میموری (RAM) بھی کہا جاتا ہے، کمپیوٹر کا بنیادی ورکنگ ایریا ہے۔ یہ ایک تیز رفتار اسٹوریج ایریا ہے جہاں کمپیوٹر ڈیٹا اور ہدایات کو عارضی طور پر محفوظ کرتا ہے۔ جب کمپیوٹر بند کر دیا جاتا ہے تو اس کا ڈیٹا ختم ہو جاتا ہے۔

مثال: جب آپ ایک Word دستاویز کھولتے ہیں، تو کمپیوٹر تیزی سے پروگرام اور دستاویز کو RAM میں رکھ دیتا ہے تاکہ CPU ان پر فوری طور پر کام کر سکے۔

شکل: 1.2 RAM

NOT FOR SALE-PESRP

ورچوئل میموری (Virtual Memory)

جب RAM بھر جاتی ہے، تو آپریٹنگ سسٹم کمپیوٹر کی اسٹوریج ڈرائیو (جیسے ہارڈ ڈرائیوز، سالڈ-اسٹیٹ ڈرائیوز (SSD)، اور Non-Volatile Memory (NVMe)) کا کچھ حصہ ورچوئل میموری (Virtual Memory) کے طور پر استعمال کرتا ہے۔ یہ اضافی جگہ عارضی RAM کی طرح کام کرتی ہے، جو زیادہ پروگراموں کو ایک ہی وقت میں چلنے کی اجازت دیتی ہے۔

اسٹوریج ڈرائیوز RAM کے مقابلے میں کم ڈیٹا کی منتقلی کی رفتار اور زیادہ رسائی کے اوقات پر کام کرتی ہیں؛ اسی وجہ سے ورچوئل میموری کا استعمال سسٹم کی کارکردگی میں کمی کا سبب بن سکتا ہے۔ جب کئی پروگرام ایک ساتھ کھلے ہوتے ہیں، تو آپریٹنگ سسٹم کم فعال پروگراموں سے ڈیٹا کو ورچوئل میموری میں منتقل کر سکتا ہے۔ یہ ان پروگرامز کے لیے RAM میں جگہ خالی کرتا ہے جو آپ فی الحال استعمال کر رہے ہیں، لیکن ان کم فعال پروگرامز کو جواب دینے میں سست کر سکتا ہے۔

کیا آپ جانتے ہیں؟

پہلا پرسنل کمپیوٹر، IBM 5150 (1981) صرف 16 KB RAM سے لیس تھا۔ اس کے برعکس، جدید DDR5 RAM 50GB, فی سیکنڈ سے زیادہ کی رفتار سے ڈیٹا منتقل کر سکتی ہے، جس سے ایک مکمل ہائی-ڈیفینیشن مووی کو ایک سیکنڈ سے بھی کم وقت میں کاپی کیا جا سکتا ہے۔

1.5 تھریڈز vs پروسیسز (Threads vs. Processes)

پروسیس (Process): ایک ایسا پروگرام ہے جس پر فی الحال کمپیوٹر کے ذریعے عملدرآمد کیا جا رہا ہے۔ اس کی اپنی میموری کی جگہ، CPU کا وقت، اور دیگر وسائل (جیسے فائلیں یا نیٹ ورک کنکشن) ہوتے ہیں۔ پروسیسز استحکام اور سیکورٹی کو یقینی بنانے کے لیے ایک دوسرے سے الگ تھلگ ہوتے ہیں؛ ایک پروسیس میں مسئلہ عام طور پر دوسرے کو متاثر نہیں کرتا۔

تھریڈ (Thread): دوسری طرف تھریڈ ایک پروسیس کے اندر عملدرآمد کی سب سے چھوٹی اکائی ہے۔ ایک ہی پروسیس کے اندر متعدد تھریڈز موجود ہو سکتے ہیں، ہر ایک مختلف کام انجام دے رہا ہوتا ہے۔ ایک ہی پروسیس میں تمام تھریڈز ایک ہی میموری اور وسائل کو شیئر کرتے ہیں، لیکن آزادانہ طور پر کام کرتے ہیں۔

مثال: ایک ویب براؤزر پر غور کریں:

- پورا براؤزر ایپلیکیشن ایک پروسیس ہے۔
- اس پروسیس کے اندر، مختلف تھریڈز اس کے ذمہ دار ہیں:
- ایک تھریڈ ایک ویب پیج کو لوڈ اور رینڈر (تصویر کشی) کر رہا ہے۔
- دوسرا تھریڈ آڈیو یا ویڈیو مواد چلا رہا ہے۔
- ایک اور تھریڈ پس منظر میں فائلیں ڈاؤن لوڈ کر رہا ہے۔
- اوپر دیئے گئے متعدد تھریڈز کا استعمال کرتے ہوئے، براؤزر ویڈیو چلاتے ہوئے بھی نیا مواد لوڈ کرنا جاری رکھ سکتا ہے

Web Version of PECTAA Textbook
Not for Sale

NOT FOR SALE-PESRP

- صارف ایک کام کے ختم ہونے کا انتظار کیے بغیر دوسرا کام شروع کر سکتا ہے۔

ملٹی تھریڈنگ

ملٹی تھریڈنگ ایک آپریٹنگ سسٹم کی تکنیک ہے جو ایک ہی پروسیس کو اپنے کام کو تھریڈز نامی چھوٹی اکائیوں میں تقسیم کر کے ایک ہی وقت میں متعدد کام انجام دینے کی اجازت دیتی ہے۔ ہر تھریڈ آزادانہ طور پر چلتا ہے لیکن پروسیس کے ہی میموری اور وسائل کو شیئر کرتا ہے، جو تیز رفتار عملدرآمد، بہتر جوابدہی، اور سسٹم کے وسائل کے موثر استعمال کو ممکن بناتا ہے۔

ملٹی تھریڈنگ کے فوائد: ملٹی تھریڈنگ کئی فوائد پیش کرتی ہے، جن میں سے کچھ اہم ذیل میں زیر بحث ہیں:

- **بہتر کارکردگی (Enhanced Performance):**

کاموں کو متعدد تھریڈز میں تقسیم کیا جا سکتا ہے اور متوازی طور پر عملدرآمد کیا جا سکتا ہے، جس سے پیچیدہ آپریشنز تیزی سے مکمل ہو سکتے ہیں اور مجموعی سسٹم کی کارکردگی میں بہتری آتی ہے۔

- **بہتر ردعمل (Improved Responsiveness):**

جب ایک تھریڈ کسی مخصوص کام میں مصروف ہوتا ہے تب بھی ایپلی کیشنز ردعمل دیتی رہتی ہیں۔

مثال: ایک ورڈ پروسیسر مسلسل ٹائپنگ کو ممکن بناتا ہے جبکہ دوسرا تھریڈ پس منظر میں ہجے کی جانچ (Spell Checking) کرتا ہے۔

- **ہم آہنگ (متوازی) آپریشنز کے لیے سپورٹ (Support for Concurrent Parallel Operations):**

متعدد کام ایک ہی مدت کے دوران ہو سکتے ہیں۔ جو خاص طور پر گیمز، ویڈیو ایڈیٹنگ، اور ریئل ٹائم مواصلاتی ٹولز جیسی ایپلی کیشنز میں اہم ہے۔

- **وسائل کا موثر استعمال (Efficient Use of Resources):**

تھریڈز اپنے پیرنٹ پروسیس کی ہی میموری اور وسائل کو شیئر کرتے ہیں، جس سے الگ پروسیسز بنانے کے مقابلے میں سسٹم کے وسائل کی ضرورت کم ہوتی ہے۔

1.6 سسٹم کالز

سسٹم کال ایک درخواست ہے جو کسی پروگرام کے ذریعے آپریٹنگ سسٹم سے ایک مخصوص کام انجام دینے کے لیے کی جاتی ہے جو پروگرام براہ راست نہیں کر سکتا۔ وہ صارف کے پروگرامنگ اور کرنل (kernel) کے درمیان ایک پل کا کام کرتی ہیں، جس سے ایپلی کیشنز کو ہارڈ ویئر اور آپریٹنگ سسٹم کے بنیادی افعال تک محفوظ طریقے سے رسائی حاصل کرنے کی اجازت ملتی ہے۔ سسٹم کالز کے بغیر، وہ پروگرام جو براہ راست ہارڈ ویئر کو کنٹرول کرتے ہیں، غیر محفوظ اور پیچیدہ ہو سکتے ہیں۔

مثال: جب آپ ایک ٹیکسٹ ایڈیٹر میں کوئی فائل محفوظ کرتے ہیں، تو پروگرام آپریٹنگ سسٹم کو اسٹوریج ڈرائیو (جیسے ہارڈ ڈرائیو) پر ڈیٹا لکھنے کے لیے کہنے کے لیے ایک سسٹم کال کا استعمال کرتا ہے۔

Web version of PECTAA Textbook Not for Sale

سسٹم کالز کی اقسام

سسٹم کالز کی اہم اقسام میں شامل ہیں:

1. open: پڑھنے یا لکھنے کے لیے ایک فائل کھولتا ہے۔
   - مثال: چلانے کے لیے ایک میوزک فائل کھولنا۔
2. read: ایک فائل یا ان پٹ ڈیوائس سے ڈیٹا حاصل کرتا ہے۔
   - مثال: کی بورڈ سے ان پٹ لینا
3. write: ایک فائل یا آؤٹ پٹ ڈیوائس کو ڈیٹا بھیجتا ہے۔
   - مثال: LED/LCD پر ایک تصویر دیکھنا۔
4. fork: ایک موجودہ پروسیس کی نقل کر کے ایک نیا پروسیس بناتا ہے۔
   - مثال: ایک نیا براؤزر ٹیب کھولنا، جہاں OS ایک اور پروسیس بنانے کے لیے fork کا استعمال کرتا ہے۔

کیا آپ جانتے ہیں؟

لینکس اور میک او ایس میں چند سو سسٹم کالز ہیں، جبکہ ونڈوز تقریباً 2,000 سسٹم کالز استعمال کرتا ہے۔ یہ کالز فائلوں کو کھولنے سے لے کر ایپس کو چلانے اور آپ کے کام کے دوران پس منظر میں گرافکس دکھانے تک سب کچھ سنبھالتی ہیں۔

1.7 فائل سسٹم کی ساخت اور انتظام (File System Structure and Management)

ایک آپریٹنگ سسٹم نہ صرف پروگرام چلاتا ہے بلکہ صارفین کے ڈیٹا کو بھی محفوظ اور منظم کرتا ہے۔ یہ فائل سسٹم کے ذریعے حاصل کیا جاتا ہے، جو صارفین اور ایپلی کیشنز کو اسٹوریج ڈیوائسز پر معلومات کو ذخیرہ کرنے، تلاش کرنے اور منظم کرنے کا ایک منظم طریقہ فراہم کرتا ہے۔ ایک اچھی طرح سے ڈیزائن کیا گیا فائل سسٹم اس بات کو یقینی بناتا ہے کہ ڈیٹا منظم، محفوظ، اور آسانی سے قابل بازیافت رہے، یہاں تک کہ جب ہزاروں فائلیں پیچیدہ ڈیٹا پر مشتمل ہوں۔

فائلیں (Files)

فائل کمپیوٹر پر ذخیرہ شدہ متعلقہ ڈیٹا کا ایک مجموعہ ہے۔ اس میں ٹیکسٹ، تصاویر، آڈیو، ویڈیو، یا پروگرام کی ہدایات شامل ہو سکتی ہیں۔

فولڈرز (Folders)

ایک فولڈر (جسے ڈائریکٹری بھی کہا جاتا ہے) ایک کنٹینر ہے جو فائلوں اور دیگر فولڈرز کو ایک منطقی ساخت میں منظم کرنے کے لیے استعمال ہوتا ہے، جس سے معلومات کو تلاش کرنا اور منظم کرنا آسان ہو جاتا ہے۔

Web version of PECTAA Textbook Not for Sale

NOT FOR SALE-PESRP

میٹاڈیٹا (Metadata)

میٹاڈیٹا سے مراد ایک فائل کے بارے میں تفصیلات ہے، جیسے اس کا نام، قسم، سائز، تخلیق کی تاریخ، اور آخری ترمیم کی تاریخ۔ یہ معلومات آپریٹنگ سسٹم اور صارف دونوں کو فائلوں کو کھولے بغیر ان کی شناخت اور منظم کرنے میں مدد کرتی ہیں۔

فائل سسٹمز (File Systems)

ایک فائل سسٹم وہ طریقہ ہے جس سے ایک آپریٹنگ سسٹم ایک اسٹوریج ڈیوائس، جیسے ہارڈ ڈرائیو، SSD، یا USB پر فائلوں کو ذخیرہ اور منظم کرتا ہے۔ یہ فیصلہ کرتا ہے کہ ہر فائل کہاں رکھی جائے گی اور اس کے مقام کو یاد رکھتا ہے تاکہ بعد میں اس تک رسائی حاصل کی جا سکے۔

کچھ بڑے پیمانے پر استعمال ہونے والے فائل سسٹمز میں درج ذیل شامل ہیں:

- FAT 32: USB فلیش ڈرائیوز میں استعمال ہوتا ہے۔
- NTFS: ونڈوز OS میں استعمال ہوتا ہے۔
- APFS / HFS+ :macOS میں استعمال ہوتا ہے۔
- EXT4: عام طور پر لینکس OS کے لیے استعمال ہوتا ہے۔

فائل سسٹم کا انتخاب کارکردگی، اسٹوریج کی صلاحیت اور سیکیورٹی خصوصیات کا تعین کرتا ہے۔

مثال: اگر آپ اپنے کمپیوٹر پر ایک تصویر محفوظ کرتے ہیں، تو فائل سسٹم یقینی بناتا ہے کہ یہ صحیح جگہ پر ذخیرہ ہو اور جب آپ اسے دوبارہ کھولتے ہیں تو اسے کہاں تلاش کرنا ہے، یہ جانتا ہے، ایک لائبریرین کی طرح جو ایک کتاب کو صحیح سیکشن میں رکھتا ہے اور جانتا ہے کہ بعد میں اسے کہاں تلاش کرنا ہے۔

سرگرمی:

مقصد: مظاہرہ کریں کہ فائلز کا انتظام کرتے وقت سسٹم کالز کیسے کام کرتی ہیں۔

ہدایات:

تیاری: ہر طالب علم کو ایک فائل کے نام کا لیبل لگا ہوا ایک چھوٹا کارڈ دیں۔ (مثلاً photo.jpg، homework.docx، song.mp3)

تختہ سیاہ (board) پر open، read، write، اور close لکھیں۔

طریقہ کار:

1. ایک طالب علم کو آپریٹنگ سسٹم کہیں۔
2. ایک اور طالب علم ایپلیکیشن کا کردار ادا کرتا ہے (مثلاً، فوٹو ویور)۔
3. ایپلیکیشن درخواست کرتی ہے: "OS، براہ کرم photo.jpg کھولیں"۔
4. OS فائل کی فہرست سے کارڈ منتخب کر کے اسے طالب علم کے حوالے کر کے جواب دیتا ہے۔
5. مختلف فائلوں کے لیے read، write، اور close کے ساتھ دہرائیں۔

بحث کا اختتام: وضاحت کریں کہ ہر عمل ایک سسٹم کال کی نمائندگی کرتا ہے۔

Web version of PECTAA Textbook Not for Sale

NOT FOR SALE-PESRP

1.8 آپریٹنگ سسٹمز کی اقسام

آپریٹنگ سسٹمز کو ڈیوائس کی ضروریات اور اس کے انجام دینے والے کام کے مطابق ڈیزائن کیا جاتا ہے۔ ہر قسم ایک مختلف مقصد کی تکمیل کرتی ہے اور مخصوص کاموں کے لیے موزوں ہوتی ہے۔

ریئل-ٹائم آپریٹنگ سسٹم (RTOS)

ایک ریئل-ٹائم آپریٹنگ سسٹم (RTOS) کو ایک مقررہ وقت (deadline) کے اندر ڈیٹا پر کارروائی کرنے اور جواب دینے کے لیے ڈیزائن کیا ہے۔ یہ وہاں استعمال ہوتا ہے جہاں معمولی تاخیر بھی سسٹم کی ناکامی یا سنگین نتائج کا سبب بن سکتی ہے۔

- اہم خصوصیت: کاموں کو پہنچتے ہی فوری طور پر پروسیس کرتا ہے۔
- کہاں استعمال ہوتا ہے: حساس سسٹمز جیسے ہوائی ٹریفک کنٹرول، دل کی نگرانی کرنے والے آلات، یا صنعتی روبوٹس وغیرہ۔

ایمبیڈڈ آپریٹنگ سسٹم (Embedded Operating System) (EOS)

ایک ایمبیڈڈ OS (Embedded OS) ایک چھوٹا اور انتہائی موثر آپریٹنگ سسٹم ہے جو صرف ان افعال کو کنٹرول کرنے کے لیے ایک مخصوص ڈیوائس میں بنایا جاتا ہے جن کی اسے ضرورت ہوتی ہے۔ یہ عام مقصد کی کمپیوٹنگ کے لیے نہیں ہے بلکہ ایک کام یا کاموں کے ایک چھوٹے سیٹ کے لیے موزوں ہے۔

- اہم خصوصیت: بہت کم میموری اور بجلی استعمال کرتا ہے، اور اکثر ڈیوائس کے اندر مستقل طور پر محفوظ ہوتا ہے۔
- کہاں استعمال ہوتا ہے: گھریلو آلات (مائیکروویوز، واشنگ مشینیں)، پرنٹرز، سمارٹ ٹی وی، اور ATM وغیرہ۔

نیٹ ورک آپریٹنگ سسٹم (Network Operating System) (NOS)

انیٹ ورک آپریٹنگ سسٹم (Netwrok OS) ایک نیٹ ورک کے ذریعے جڑے ہوئے متعدد کمپیوٹرز کو منظم اور سپورٹ کرتا ہے۔ یہ صارفین کے درمیان فائلز، پرنٹرز، اور انٹرنیٹ کنکشنز جیسے وسائل کو شیئر کرواتا ہے۔ (جیسے ونڈوز ملٹی پوائنٹ سرورز)۔

- اہم خصوصیت: کمپیوٹرز کے درمیان مواصلات اور ہم آہنگی پر توجہ مرکوز کرتا ہے۔
- کہاں استعمال ہوتا ہے: دفاتر، اسکول، اور ڈیٹا سینٹرز۔

مثال: اسکول کی کمپیوٹر لیب میں، طلباء مختلف کمپیوٹر استعمال کرتے ہیں، لیکن وہ سب ایک ہی سرور پر فائلیں محفوظ کر سکتے ہیں اور ایک ہی پرنٹر سے پرنٹ کر سکتے ہیں۔

موبائل آپریٹنگ سسٹم (Mobile Operating System)

ایک موبائل (Mobile OS) سمارٹ فونز، ٹیبلٹس، اور دیگر ہینڈ ہیلڈ ڈیوائسز کے لیے ڈیزائن کیا گیا ہے۔ یہ ٹچ-اسکرین کے استعمال، بیٹری کی بچت، اور موبائل ایپس کے لیے موزوں ہے۔

Web version of PECTAA Textbook Not for Sale

- اہم خصوصیت : وائرلیس کنیکٹیوٹی، کیمرے، سینسرز، اور ایپ اسٹورز کو سپورٹ کرتا ہے۔
- کہاں استعمال ہوتا ہے : سمارٹ فونز، ٹیبلٹس، سمارٹ واچز۔

کیا آپ جانتے ہیں؟

سب سے پہلے موبائل آپریٹنگ سسٹمز، جیسے Symbian (جو ابتدائی نوکیا فونز میں استعمال ہوتا تھا)، صرف چند چھوٹی ایپس چلا سکتے تھے۔ آج کے موبائل OS پلیٹ فارمز جیسے اینڈرائیڈ اور آئی او ایس لاکھوں ایپس، تھری ڈی گیمز، اور یہاں تک کہ پیشہ ورانہ ویڈیو ایڈیٹنگ ٹولز کو بھی سنبھال سکتے ہیں۔

خلاصہ (Summary)

- آپریٹنگ سسٹم (OS) ایک قسم کا سسٹم سافٹ ویئر ہے جو ہارڈ ویئر کا انتظام کرتا ہے، ایپلی کیشن سافٹ ویئر چلاتا ہے، اور صارف کو انٹرفیس فراہم کرتا ہے۔
- کرنل (Kernel) آپریٹنگ سسٹم کا مرکزی حصہ ہے، جو براہِ راست کمپیوٹر کے ہارڈ ویئر اور سسٹم سافٹ ویئر جیسے کہ CPU، میموری اور دیگر آلات کو کنٹرول کرتا ہے۔
- شیل (Shell) آپریٹنگ سسٹم کا بیرونی حصہ ہے جو صارف کے ساتھ رابطہ (interact) کرتا ہے۔
- سسٹم لائبریریاں تیار شدہ ہدایات کے مجموعے ہیں جنہیں پروگرام عام کام انجام دینے کے لیے استعمال کر سکتے ہیں، جیسے فائلیں کھولنا یا اسکرین پر تحریر دکھانا۔
- آپریٹنگ سسٹم کمپیوٹر پر چلنے والے تمام پروگراموں کے انتظام کا ذمہ دار ہے۔ اس تناظر میں، ان چلنے والے پروگراموں کو پراسیس کہا جاتا ہے۔
- ملٹی ٹاسکنگ میں، آپریٹنگ سسٹم ایک ہی وقت میں ایک ہی صارف کو ایک سے زیادہ پروگرام کھولنے اور استعمال کرنے کی اجازت دیتا ہے۔
- کانکورنسی (Concurrency) میں، آپریٹنگ سسٹم میں ایک ہی وقت میں ایک سے زیادہ پراسیس فعال ہوتے ہیں، لیکن CPU انہیں انتہائی تیز رفتار چکروں میں ایک ایک کرکے پروسیس کرتا ہے۔
- FCFS طریقہ کار میں، CPU کاموں کو اسی ترتیب سے پروسیس کرتا ہے جس ترتیب سے وہ آتے ہیں۔ سب سے پہلے آنے والا پراسیس پہلے مکمل ہوتا ہے، اور اگلا پراسیس پچھلے کے ختم ہونے کے بعد ہی شروع ہوتا ہے۔
- جب RAM بھر جاتی ہے، تو آپریٹنگ سسٹم کمپیوٹر کے اسٹوریج ڈرائیو جیسے ہارڈ ڈرائیوز، سالڈ اسٹیٹ ڈرائیوز (SSD)، اور Non Volatile Memory Express (NVMe) کے کچھ حصے کو ورچوئل میموری کے طور پر استعمال کرتا ہے۔
- پراسیس ایک آزاد پروگرام ہے جو اس وقت کمپیوٹر کے ذریعے چلایا جا رہا ہوتا ہے۔ دوسری طرف، تھریڈ (thread) ایک پراسیس کے اندر عملدرآمد کی سب سے چھوٹی اکائی ہے۔
- سسٹم کال (System call) ایک ایسی درخواست ہے جو ایک پروگرام کی طرف سے آپریٹنگ سسٹم کو کسی خاص کام کو انجام دینے کے لیے دی جاتی ہے جسے پروگرام براہِ راست نہیں کر سکتا۔

Web version of PECTAA Textbook
Not for Sale

NOT FOR SALE-PESRP

کثیر انتخابی سوالات

i درج ذیل میں سے کون سی چیز آپریٹنگ سسٹم کی مثال نہیں ہے؟

(الف) لینکس(Linux)
(ب) ونڈوز(Windows)
(ج) فوٹوشاپ(Photoshop)
(د) میک او ایس(macOS)

ii ملٹی یوزر ماحول میں، OS اس بات کو یقینی بناتا ہے:

(الف) مساوی رسائی اور ڈیٹا کی رازداری(Equal access and data privacy)
(ب) تمام صارفین ایک ہی فائلیں شیئر کرتے ہیں
(ج) ہارڈویئر کبھی استعمال نہیں ہوتا
(د) انٹرنیٹ کی رفتار بڑھ جاتی ہے

iii ونڈوز میں، یوزر اکاؤنٹس کس کے ذریعے بنائے جاسکتے ہیں؟

(الف) ٹاسک مینیجر(Task Manager)
(ب) کنٹرول پینل یا سیٹنگز(Control Panel or Settings)
(ج) ڈسک مینجمنٹ(Disk Management)
(د) BIOS سیٹنگز

iv OS کا بنیادی حصہ جو براہ راست ہارڈویئر کے ساتھ تعامل کرتا ہے، وہ ہے:

(الف) شیل(Shell)
(ب) کرنل(Kernel)
(ج) ڈیوائس ڈرائیور(Device Driver)
(د) سسٹم لائبریری(System Library)

v گرافیکل شیل یوزر کو اجازت دیتا ہے:

(الف) صرف کمانڈ ٹائپ کرنا
(ب) آئیکن پر کلک کرنا اور مینوز کا استعمال کرنا (Click icon and use menus)
(ج) صرف کوڈ کے ذریعے رابطہ کرنا
(د) صرف BIOS تک رسائی حاصل کرنا

Web version of PECTAA Textbook
Not for Sale

NOT FOR SALE-PESRP

vi FCFS شیڈولنگ طریقہ کار میں، پروسیسز کو سرو کیا جاتا ہے:

(الف) بے ترتیب (Randomly)
(ب) سب سے پہلے مختصر کام
(ج) اسی ترتیب میں جس میں وہ پہنچتے ہیں (In the order they arrive)
(د) صرف ترجیح (priority) کے لحاظ سے

vii ایک ہی پروسیس میں تھریڈز:

(الف) الگ میموری کی جگہ رکھتے ہیں
(ب) ایک ہی میموری اور وسائل کو شیئر کرتے ہیں۔
(ج) مختلف OS پر چلتے ہیں
(د) ایک ساتھ نہیں چل سکتے

viii نیا پروسیس بنانے کے لیے کون سا سسٹم کال استعمال ہوتا ہے؟

(الف) اوپن
(ب) ریڈ (Read)
(ج) رائٹ (Write)
(د) فورک

مختصر سوالات

1. آپریٹنگ سسٹم کو مرکزی کنٹرولر کیوں کہا جاتا ہے؟
2. OS صارف اور ہارڈویئر کے درمیان مترجم کے طور پر کیسے کام کرتا ہے؟
3. ونڈوز میں نیا یوزر اکاؤنٹ بنانے کا ایک طریقہ بیان کریں۔
4. کرنل شیل سے کیسے مختلف ہے؟
5. ایک سسٹم لائبریری اور اس کے مقصد کی ایک مثال دیں۔
6. FCFS شیڈولنگ کا ایک فائدہ اور ایک نقصان بیان کریں۔
7. ورچوئل میموری RAM سے سست کیوں ہے؟
8. سسٹم کال کی تعریف کریں۔
9. فائل سسٹم کا کردار کیا ہے؟
10. کمپیوٹر سسٹم میں ملٹی ٹاسکنگ کی تعریف کریں۔

تفصیلی سوالات

1. آپریٹنگ سسٹم کے آرکیٹیکچر کو بیان کریں، جس میں کرنل، شیل، اور لیئرڈ ڈیزائن کی مثال کے ساتھ وضاحت کی جائے۔
2. پروسیس لائف سائیکل کی مثال کے ساتھ تفصیل سے وضاحت کریں۔
3. RAM اور ورچوئل میموری کے درمیان فرق واضح کریں، اور وضاحت کریں کہ ملٹی تھریڈنگ کارکردگی کو کیسے بہتر بناتی ہے؟
4. سسٹم کالز کیا ہیں؟ ان کے مقصد کو بیان کریں، اور کم از کم تین اقسام کی مثالیں دیں۔

Web version of PECTAA Textbook
Not for Sale

NOT FOR SALE-PESRP

5. ایک کمپیوٹر سسٹم پروسیسز کو منظم کرنے کے لیے فرسٹ-کم، فرسٹ-سروڈ(FCFS) شیڈولنگ طریقہ کار استعمال کرتا ہے۔
درج ذیل ٹیبل میں تین پانچ پروسیسز کی آمد کا وقت اور CPU کے کام کرنے (Burst) کا وقت (عملدرآمد کے لیے درکار وقت) دکھایا گیا ہے:

برسٹ کا وقت (Burst Time - seconds)	آمد کا وقت (Arrival Time - seconds)	پروسیس (Process)
4	0	P1
3	1	P2
1	2	P3

ضروریات (Requirements):

- FCFS شیڈولنگ طریقہ کار کے مطابق پروسیسز کو اس ترتیب میں لکھیں جس میں وہ چلائے جائیں گے۔
- تمام پروسیسز کی عملدرآمد کی ترتیب دکھائیں۔
- ہر پروسیس کے لیے شروع ہونے کا وقت اور تکمیل کے وقت کا حساب لگائیں۔
- ہر پروسیس کے لیے انتظار کا وقت (Waiting Time) معلوم کریں۔
- تمام پروسیسز کے لیے اوسط انتظار کا وقت (Average Waiting Time) کا حساب لگائیں۔

Web version of PECTAA Textbook Not for Sale

کثیر انتخابی سوالات کے جوابات

1. ج فوٹوشاپ (Photoshop)
2. الف مساوی رسائی اور ڈیٹا کی رازداری (Equal access and data privacy)
3. ب کنٹرول پینل یا سیٹنگز (Control Panel or Settings)
4. ب کرنل (Kernel)
5. ب آئیکنز پر کلک کرنا اور مینیوز کا استعمال کرنا (Click icons and use menus)
6. ج اسی ترتیب میں جس میں وہ پہنچتے ہیں (In the order they arrive)
7. ب ایک ہی میموری اور وسائل کو شیئر کرتے ہیں (Share the same memory and resources)
8. د fork

NOT FOR SALE-PESRP