Chapter５

ソフトウェア

# 1. OS

## 1. OSの種類と特徴

学習のポイント

✅ OSの基礎として確認しておこう！

コンピュータを動かすにはソフトウェアが必要です。ソフトウェアには大きく分けるとシステムソフトウェアと応用ソフトウェア（アプリケーションソフトウェア）があります。

システムソフトウェアは、パソコンの基本的な機能を働かせるために必要なソフトウェアです。OS（Operating System：オペレーティングシステム）や言語プロセッサなどのハードウェアやデータ資源などを管理する基本ソフトウェアと呼ばれるソフトウェア群と、基本ソフトウェアの機能を利用して、多様な利用分野で共通するより高いレベルの基本機能を提供するミドルウェアと呼ばれるソフトウェア群から構成されています。

応用ソフトウェアは、表や文書の作成、給与計算など特定の目的のために作成されたソフトウェアです。

これらのソフトウェアがハードウェアを操ってコンピュータを動かしています。

ユーザ

アプリケーション

オペレーティングシステム

ハードウェア

ミドルウェア

ソフトウェアの階層構造

OSには、パソコン用に開発されたWindowsやMac OS、ワークステーション用に開発されたUNIX、汎用コンピュータ用に開発されたMVSなどの情報処理用のOSと、TRONのように機器の制御を目的として開発されたOSがあります。後者をリアルタイムOSと呼びます。

リアルタイムOSには、一定時間内に処理が実行されるようにさまざまな機能が実装されています。

|  |
| --- |
| 例題  エンジン制御，ハードディスク制御などの制御系ハードリアルタイムシステムでリアルタイムOSを活用する理由はどれか。  ア　ウイルス侵入の防御のためにメモリ保護機構が必要だから。  イ　定められた時間内にイベントに対応した処理を完了させる機構が必要だから。  ウ　システム全体のスループットを向上させる機構が必要だから。  エ　複数の独立したプログラムを並行して動かす機構が必要だから。  ハードリアルタイムシステムは、時間内に処理を完了させないと致命的な事態が発生してしまいます。そのため、限られた時間内に処理を完了させる機構であるリアルタイムOSを活用する必要があります。  基本情報　平成22年度春　問18　[出題頻度：★☆☆]  解答－イ |

## 2. OSの機能と構成

学習のポイント

✅ OSには多くの機能が含まれていることを知っておこう！

OSは、CPUやメモリ、補助記憶装置などのコンピュータ資源を管理し、利用者や応用ソフトウェアに対して、さまざまなサービスを提供するソフトウェアです。

OSは、次に示すソフトウェア群から構成されており、ジョブ管理、タスク管理、記憶管理、データ管理、入出力管理などの機能を有しています。

なお、OSの種類によって利用できる応用ソフトウェアは限られます。

…

(ユーティリティプログラム)

連係編集プログラム

ローダ

デバッキングエイド

テキスト編集プログラム

言語プロセッサ

アセンブラ

コンパイラ

ジェネレータ

インタプリタ

…

サービスプログラム

デバイスドライバ

ファイルシステム

（狭義のOS）

（広義のOS）

基本ソフトウェア

制御プログラム

カーネル

ジョブ管理

主記憶管理

データ管理

タスク管理

プログラム管理

多重割込み管理

入出力管理

カーネル処理管理

システムコールサービス

OSの構成

🏋プラスアルファ

**●マイクロカーネルとモノリシックカーネル**

ジョブ管理やタスク（プロセス）管理、主記憶管理など、主記憶上に常駐し、CPUに最も近い処理を行うプログラム群を**カーネル**（kernel）と呼びます。

なお、カーネルには最小限の機能しかもたせず、なるべく多くの機能を外部モジュールとして提供する設計手法をマイクロカーネルと呼び、逆に、カーネルにさまざまな機能を実装する設計手法をモノリシックカーネルと呼びます。

|  |
| --- |
| 例題　プラスアルファ  OSが提供する機能を最小限のメモリ管理やプロセス管理などに限定し，ファイルシステムなど他のOS機能はサーバプロセスとして実現されているものはどれか。  ア　シングルユーザモード イ　マイクロカーネル  ウ　マルチスレッド エ　モノリシックカーネル  OSの中核部分（カーネル）には最小限の機能しかもたせず、なるべく多くの機能を外部モジュールとして提供する設計手法をマイクロカーネルといいます。  ア　マルチユーザ環境では、常に複数のユーザがコンピュータの資源を使用しているため、システム管理用ファイルの書換えやファイルシステムのバックアップ、アカウントファイル（コマンドの履歴及びユーザの利用状況）の厳密な操作を行うには、システム管理者のみが作業を行っている状態が必要になります。このためのモードがシングルユーザモードです。  ウ　同一プロセス内でのマルチタスク処理を可能にしたものがマルチスレッドシステムです。なお、この場合のタスクの実行の単位をスレッドと呼びます。  エ　カーネルに様々な機能を実装する手法によって設計されたカーネルをモノリシックカーネルといいます。現在、ほとんどのOSはモノリシックカーネルを採用しています。  基本情報　平成17年度秋　問31　[出題頻度：★☆☆]  解答－イ |

🏋プラスアルファ

**●BIOS**（Basic Input/Output System：バイオス）

BIOSは、コンピュータに接続された周辺装置を制御するためのプログラム群です。

コンピュータに電源が入れられると、内部にあらかじめ搭載されているROMに記録されたBIOSが動作して、主記憶をクリアし、ハードディスクなどの補助記憶装置、キーボードやディスプレイなど基本的な入出力装置を使用可能な状態にします。その後、マザーボードに記憶されたIPL（Initial Program Loader）が起動して、ハードディスクからOSをメモリに読み込むことでOSが起動します。この一連の動作をブートストラップと呼んでいます。なお、OSが起動すると、これらの周辺装置の管理はOSが行うことになります。

## 3. ジョブ管理

学習のポイント

✅ ジョブスケジューリングの種類と特徴を覚えよう！

**ジョブ**とは、利用者から見たひとまとまりの仕事の単位です。通常、コンピュータに１つの処理をさせる場合、データの準備、実行など複数の動作が必要であり、この一連の動作を１つのジョブといい、ジョブを構成する各動作（プログラム）を**ジョブステップ**といいます。

ジョブ管理は、ジョブの連続実行と、これにかかわる資源の有効利用を目的とし、使用する資源やジョブの実行手順などは、ジョブ制御言語（JCL：Job Control Language）で記述します。

例　ジョブ（ジョブステップ）の実行の流れ

『大宮支店の1年間の売上金額を印刷するプログラム』

・ジョブの連続実行

　ジョブステップの間に、人間による手作業が発生する  
　　と処理装置の使用効率が低下してしまう。

全店の売上データ

大宮支店の

売上データ

整列済み

売上データ

大宮支店売上一覧表

1月　　　　￥585、000

2月　　　　￥423、000

…

全店のデータより大宮支店の  
データを抽出する。

抽出したデータを各月ごとに  
整列させる。

整列されたデータを、各月ごとに印刷させる。

ジョブステップ２

ジョブステップ１

ジョブステップ３

ジョブの連続実行

　そこで、OS（ジョブ管理プログラム）を導入すること  
　　により、ジョブステップの自動連続実行を実現し、シス  
　　テム全体の処理効率を高める。

・ジョブ制御言語（JCL）

　複数のジョブステップをどのような順番で実行させるか、  
　　コンピュータに指示させる必要があり、この指示をする  
　　ために記述した言語をいう。

ジョブ管理の目的

🏋プラスアルファ

**●ジョブ管理の主要な機能**

**１）ジョブスケジューラの機能**

ジョブスケジューラは、ジョブの連続実行を実現するための機能であり、次の４つのプログラムから構成されています。

**①リーダ（ジョブ読取りプログラム）**

リーダは、ジョブを入力し、入力ジョブ待ち行列に登録します。

**②イニシエータ（ジョブ開始プログラム）**

イニシエータは、入力ジョブ待ち行列に登録されているジョブの中から優先順位の高いジョブを選択し、そのジョブを実行可能な状  
　　態にします。

**③ターミネータ(ジョブ終止プログラム）**

ターミネータは、実行が終了したジョブに割り当てられていた資源を解放するとともに、結果を出力ジョブ待ち行列に登録します。

**④ライタ（ジョブ書出しプログラム）**

ライタは、出力ジョブ待ち行列に登録されているジョブの中から優先順位の高いジョブを選択し、そのジョブを出力します。

**２）マスタスケジューラの機能**

マスタスケジューラは、コンピュータと利用者の間で連絡を行う機能です。

入力ジョブ

待ち行列

出力ジョブ

待ち行列

イニシエータ

入力装置

リーダ

実行

ターミネータ

ライタ

データ

出力装置

マスタスケジューラ

オペレータ

ジョブ管理機能

**３）ジョブスケジューリング**

システム全体の効率を高めるために、ジョブの処理順序を決めることをジョブスケジューリングといいます。

**①FCFS方式（到着順方式）**

FCFS（First Come First Served）方式は、ジョブを先着順に処理する方式で、到着順方式とも呼ばれます。  
　　実行中のジョブが終了するまで、後続のジョブ処理が待たされるという問題点があります。

**②ラウンドロビン方式**

FCFS方式の問題点を解決するために、CPUを使用できる制限時間（タイムクォンタム：time quantum）を決める時分割  
　　方式が考え出されました。

ラウンドロビン方式は、時分割方式におけるジョブのスケジューリング方式で、先に到着したジョブから順に処理され、一定時間  
　　CPUが割り当てられたジョブは、割当て終了後に、実行を待つジョブの待ち行列の最後尾に回されます。

**③SJF方式**（処理時間順方式）

SJF（Shortest Job First）方式は、実行時間が短いジョブを優先的に待ち行列の先頭に配置する方式で、**SPT**（Shortest   
　　Processing Time First）**方式**や処理時間順方式とも呼ばれます。

**④優先順位**（priority scheduling）**方式**（優先度順方式）

ジョブの優先度は一律ではなく、一般にリアルタイムのジョブ、OSのジョブ、対話型のジョブ、バッチジョブの順に優先度が割り当て  
　　られています。優先順位方式は、ジョブの優先度を考慮したスケジューリング方式で、優先度が高いジョブは優先度の低いジョブよ  
　　りも優先的にCPUを使用することができます。

これは、優先度に応じた待ち行列を用意し、優先度の高い待ち行列から順にCPUを割り当てていくイメージです。ただし、この方  
　　式では、優先度が低いジョブにはなかなかCPUが割り当てられないという問題があります。この状態をスタベーション（starvation：  
　　無期延期）といい、スタベーション対策として待ち時間に比例して優先度を上げることを、エージング（aging）といいます。

**⑤多重待ち行列方式（フィードバック待ち行列方式）**

多重待ち行列方式は、優先度順方式とラウンドロビン方式を合わせた方式で、フィードバック待ち行列方式とも呼ばれます。フィー  
　　ドバックとは、以前の結果を何らかの形で現状に反映することをいいます。多重待ち行列方式では、CPUの制限時間を使い切った  
　　タスクは、優先度が低い待ち行列に並ぶことになります。つまり、実行時間の長いジョブは、優先度を下げるという形でフィードバック  
　　が行われます。

|  |
| --- |
| 例題  OSのスケジューリング方式に関する記述のうち，適切なものはどれか。  ア　処理時間順方式では，既に消費したCPU時間の長いジョブに高い優先度を与える。  イ　到着順方式では，ラウンドロビン方式に比べて特に処理時間の短いジョブの応答時間が短くなる。  ウ　優先度順方式では，一部のジョブの応答時間が極端に長くなることがある。  エ　ラウンドロビン方式では，ジョブに割り当てるCPU時間（タイムクウォンタム）を短くするほど，到着順方式に近づく。  優先度順方式は、各ジョブに優先度を持たせ、その優先度にしたがって実行していく方式です。この方式では、実行中のジョブより優先度の高いジョブが実行可能になった場合、実行中のジョブは一時実行を中断し、優先度の高いジョブを実行していくことになります。これを繰り返していくと、優先度の低いジョブは、処理を終了するまでに極端に時間がかかってしまうことがあります。  ア　処理時間順方式は、処理時間の短いジョブから実行していく方式です。  イ　到着順方式は、ジョブに優先順位を持たせず、実行可能になったジョブから処理する方式です。ラウンドロビン方式は、処理時間に関係なく、各ジョブに均等にCPUの使用時間を割り当てて処理する方式です。したがって到着順方式では、処理時間の長いジョブが先に到着した場合、処理時間の短いジョブの応答時間がラウンドロビン方式に比べ長くなります。  エ　タイムクウォンタムを短くすると、処理時間の短いジョブの実行が先に終了する可能性が高くなります。したがって処理時間順方式に近づきます。  応用情報　平成23年度春　問19　[出題頻度：★★☆]  解答－ウ |

別冊演習ドリル 》 1-349～1-353

## 4. タスク（プロセス）管理

学習のポイント

✅ タスクの状態遷移は、動作状態の種類と遷移する要因を覚えよう！

✅ マルチプログラミングは、例題のような問題を解くための考え方を身につけよう！

タスク管理の最大の目的は、CPUを効率的に使用することにあります。

### １）タスクと状態遷移

タスクは、コンピュータから見た仕事の単位で、この単位ごとにコンピュータ資源（おもにCPU）の割当てが行われます。タスクはメインフレーム系OSで使われる用語で、パソコンやUNIX（ユニックス）系のOSでは、タスクの代わりに**プロセス**という用語が使われます。

なお、UNIX系のOSでは、プロセスよりもさらに小さい処理単位として**スレッド**（軽量プロセス）と呼ぶ単位が使われます。プロセスが個別に資源の割当てを行うのに対し、スレッドはCPU以外の資源は個別に割り当てず、必要な資源は親のプロセスから継承し、プロセス内の他のスレッドと共有します。そのため、スレッド間でのデータ交換などにかかるオーバヘッドが小さく、より高速な処理が可能となります。

|  |
| --- |
| 例題  スレッドとは，プロセス内部に含まれている論理的な並列処理の単位である。スレッドごとに用意されるものはどれか。  ア　アドレス空間 イ　開いているファイル識別子  ウ　プロセス間の通信ポート エ　レジスタ群の退避域  スレッドはOSが１つのプログラム（プロセス）の処理を複数に分割して実行する場合の、処理の最小単位です。マルチタスクOSの場合、メモリ空間などはプロセスごとに用意されるので、プロセスの切替えには非常に大きな負荷がかかってしまいます。しかし、同一プロセス内のスレッド間では、メモリ空間などを共有することができるので、スレッド間でのデータ交換などにかかるオーバヘッドが非常に小さくなります。  マルチスレッドOSでは、プロセスを複数のスレッドに分け、このスレッドごとに実行していくため、スタック領域、レジスタ、プログラムカウンタなどがスレッドごとに用意されることになります。  ソフトウェア開発　平成15年度春　問26　[出題頻度：★★☆]  解答－エ |

別冊演習ドリル 》 1-354～356

タスクは、図のような３つの動作状態の移動（遷移）を繰り返しながら実行されます。

入出力実行完了

↓

入出力割込み

入出力動作の要求により

CPUの使用権を放棄

（入出力動作の完了待ち）

↓

SVC割込み

時間切れ

↓

タイマ割込み

タスク管理

タスクの生成

実行終了

タスクの消滅

（ディスパッチ）

CPUの割当て

タスクの状態遷移

|  |  |
| --- | --- |
| タスクの状態 | 状態の説明 |
| タスクの生成 | ジョブステップによりタスクが作られます。 |
| 実行可能状態 （READY） | 他のタスクが実行中で、CPUなどのハードウェア資源の割当てを待っている状態をいいます。  実行可能状態のタスクは、優先度に従ってCPUの使用権が割り当てられ、実行可能状態から実行状態に移動します。なお、CPUの使用権を割り当てることを**ディスパッチ**といい、この作業を行うプログラムを**ディスパッチャ**といいます。 |
| 実行状態 （RUN） | タスクがCPUを占有して、実行している状態です。  実行中のタスクに入出力命令などが出た場合、SVC割込みが発生し、実行状態から待ち状態に移動します。また、実行中のタスクより優先度の高いタスクが生成されたり、CPUの使用可能時間を使い切り、タイマ割込みが発生したりした場合、実行状態から実行可能状態に移動します。 |
| 待ち状態 （WAIT） | 入出力動作の終了待ちなどのため、実行を一時中断している状態です。  入出力動作終了の信号がCPUに送られると入出力割込みが発生し、入出力動作の終了を待っていたタスクは、待ち状態から実行可能状態に移動します。 |
| タスクの消滅 | タスクの実行が終了すると、システム資源は解放され、タスクは削除されます。 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 例題  図はプロセスの状態と遷移を表している。ａ，ｂ，ｃの状態の適切な組合せはどれか。  ａ  ｃ  ｂ  ①  ②  ③  ④  〔状態遷移の要因〕  ①実行優先度の高いほかのプロセスにCPU使用権が移された。  ②CPU使用権が与えられた。  ③入出力動作の完了を待つ。  ④入出力動作が完了した。   |  |  |  |  | | --- | --- | --- | --- | |  | ａ | ｂ | ｃ | | ア | 実行可能状態 | 実行状態 | 待ち状態 | | イ | 実行可能状態 | 待ち状態 | 実行状態 | | ウ | 実行状態 | 実行可能状態 | 待ち状態 | | エ | 実行状態 | 待ち状態 | 実行可能状態 |   プロセスの状態遷移の要因から、状態ａ、ｂ、ｃを考えます。  ａ  ｃ  ｂ  ①  ②  ③  ④  ①　優先度の高いプロセスにCPU使用権が移された。･･･プリエンプション  ②　CPU使用権が与えられた。･･･ディスパッチ  ③　入出力動作の完了を待つ。･･･SVC割込み  ④　入出力動作が完了した。･･･入出力割込み  状態遷移の要因②より、CPU使用権が与えられた状態ａは、実行状態です。また、プロセス（タスク）の実行中に入出力動作が発生した場合は、タスクは待ち状態に移されるため、状態ｃは待ち状態です。入出力動作が完了すると、要因④の入出力割込みが発生し、タスクは実行可能状態に移されるため、状態ｂは実行可能状態です。  基本情報　平成18年度秋　問29　[出題頻度：★★☆]  解答－ウ |

別冊演習ドリル 》 1-357,358

### ２）マルチ（多重）プログラミング（マルチタスク）とスケジューリング

効率よくタスクを処理するために、さまざまな方法が考えられました。

#### ①マルチプログラミング

実行中のタスクが、入出力要求などで待ち状態に遷移すると、その間を利用して別のタスクを実行する仕組みがあります。これを用いると、見かけ上は複数のプログラムが同時に動いているかのように見えるところから、この仕組みをマルチプログラミング（マルチタスク）といいます。

A

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 例題  三つのタスクの優先度と，各タスクを単独で実行した場合のCPUと入出力（I/O）装置の動作順序と処理時間は，表のとおりである。優先度方式のタスクスケジューリングを行うOSの下で，三つのタスクが同時に実行可能状態になってから，全てのタスクの実行が終了するまでの，CPUの遊休時間は何ミリ秒か。ここで，CPUは１個であり，１CPUは１コアで構成され，I/Oは競合せず，OSのオーバヘッドは考慮しないものとする。また，表中の（　）内の数字は処理時間を示すものとする。   |  |  | | --- | --- | | 優先度 | 単独実行時の動作順序と処理時間(ミリ秒) | | 高 | CPU(3) → I/O(5) → CPU(2) | | 中 | CPU(2) → I/O(6) → CPU(2) | | 低 | CPU(1) → I/O(5) → CPU(1) |   ア　２ イ　３ ウ　４ エ　５  問題文の実行条件に従い、各タスクを多重プログラミングで処理すると、次のようになります。なお、図が見やすいように、優先度高、中、低をそれぞれＡ、Ｂ、Ｃとします。また、「I/Oは競合せず」という文章から、I/Oはそれぞれ別の装置を利用していると判断します。  単位　１マス１ミリ秒  1  CPU  I/O(A)  I/O(B)  I/O(C)  A  B  C  A  A  A  B  A  B  A  C  A  B  A  B  C  A  B  C  A  B  C  B  C  B  B  C  5  15  10  上図の網掛けの部分がCPUの遊休時間であるから、合計で３ミリ秒となります。  基本情報　平成30年度秋　問16　[出題頻度：★★★]  解答－イ |

別冊演習ドリル 》 1-359～361

#### ②プリエンプティブ方式とノンプリエンプティブ方式

実行中のタスクが既定時間以上にCPUを占有したか、より優先度の高いタスクが到着した場合、実行中のタスクを一時中断して、優先度の高いタスクを実行することを**プリエンプション**（preemption）といい、この性質をもつOSをプリエンプティブなOSといいます。逆に、プリエンプションを許さないことを**ノンプリエンプション**（non-preemption）といい、この性質をもつOSをノンプリエンプティブなOSといいます。

ノンプリエンプティブなマルチタスクOSは、実行可能になった順番にタスクが実行され、タスクの実行が終了するまで、タスクの切り替えは起こりません。したがって、実行中のアプリケーションプログラムがループ（無限ループ：終わりなく繰り返される）すると、タスク自らCPUを離すことができなくなり、OSに制御を戻すことができなくなるという欠点があります。

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 例題  タスクが実行状態（RUN），実行可能状態（READY），待ち状態（WAIT）の三つの状態で管理されるリアルタイムOSにおいて，三つのタスクＡ～Ｃの状態がプリエンプティブなスケジューリングによって，図に示すとおりに遷移した。各タスクの優先度の関係のうち，適切なものはどれか。ここで，優先度の関係は“高い＞低い”で示す。  時間   |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | タスクＡ | RUN | WAIT | | READY | RUN | READY | | タスクＢ | WAIT | RUN | WAIT | RUN | WAIT | | | タスクＣ | WAIT | READY | RUN | WAIT | | RUN |   ア　タスクＡ＞タスクＢ＞タスクＣ  イ　タスクＢ＞タスクＡ＞タスクＣ  ウ　タスクＢ＞タスクＣ＞タスクＡ  エ　タスクＣ＞タスクＢ＞タスクＡ  あるタスクの実行中に、他のタスクに実行が移った場合、後者のタスクの優先順位が高いことになります。これをもとに、各タスクの優先順位を検討します。  この部分でＢ＞Ｃという  関係が分かる。  この部分でＢ＞Ａという  関係が分かる。  この部分でＣ＞Ａという  関係が分かる。   |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | タスクＡ | RUN | WAIT | | READY | RUN | READY | | タスクＢ | WAIT | RUN | WAIT | RUN | WAIT | | | タスクＣ | WAIT | READY | RUN | WAIT | | RUN |   上記の結果から、タスクＢ＞タスクＣ＞タスクＡという優先順位が分かります。  ソフトウェア開発　平成18年度秋　問27　[出題頻度：★★★]  解答－ウ |

別冊演習ドリル 》 1-362～365

#### ③タスクスケジューリング

タスクの応答性やシステム全体の効率を高めるために、タスクの処理順序を決めることをタスクスケジューリングといいます。ジョブスケジューリングの方法と同様に、①FCFS方式（到着順方式）、②ラウンドロビン方式、③SJF方式（SPT方式、処理時間順方式）、④優先順位（priority scheduling）方式（優先度順方式）、⑤多重待ち行列方式（フィードバック待ち行列方式）があります。

|  |
| --- |
| 例題  スケジューリングに関する記述のうち，ラウンドロビン方式の説明として，適切なものはどれか。  ア　各タスクに，均等にCPU時間を割り当てて実行させる方式である。  [イ](https://www.fe-siken.com/kakomon/30_aki/q18.html#ans)各タスクに，ターンアラウンドタイムに比例したCPU時間を割り当てて実行させる方式である。  [ウ](https://www.fe-siken.com/kakomon/30_aki/q18.html#ans)各タスクの実行イベント発生に応じて，リアルタイムに実行させる方式である。  [エ](https://www.fe-siken.com/kakomon/30_aki/q18.html#ans)各タスクを，優先度の高い順に実行させる方式である。  イ　到着順方式の説明です。  ウ　イベントドリブン方式の説明です。  エ　優先順方式の説明です。  基本情報　平成30年度秋　問18　[出題頻度：★★☆]  解答－ア |

別冊演習ドリル 》 1-366,367

#### ④同期制御

並行して動作しているプロセス同士が共有する資源にアクセスする場合、処理の結果に矛盾が生じないように、あるプロセスの完了を待ってから、次のプロセスの実行を開始することがあります。そのための制御を同期制御と呼びます。

|  |
| --- |
| 例題  タスク間共有変数を更新するときに，タスク間の同期制御を行わないと思わぬ結果を招くことがある。タスク間共有変数ｘの初期値が３のとき，タスクＡが代入文ｘ＝ｘ＋ｘを実行し，タスクＢが代入文ｘ＝ｘ×ｘを実行すると，最終的なｘの値が12となるのは，どの順番で実行されたときか。ここで，各代入文はそれぞれ次の四つの部分に分けて処理される。  タスクＡ（ｘ＝ｘ＋ｘ）  a1　ｘの値を参照しｅとする。  a2　ｘの値を参照しｆとする。  a3　ｅ＋ｆを計算しｇとする。  a4　ｇによってｘの値を更新する。  タスクＢ（ｘ＝ｘ×ｘ）  b1　ｘの値を参照しｈとする。  b2　ｘの値を参照しｉとする。  b3　ｈ×ｉを計算しｊとする。  b4　ｊによってｘの値を更新する。  ア　a1 → a2 → b1 → b2 → a3 → a4 → b3 → b4  イ　a1 → b1 → b2 → b3 → b4 → a2 → a3 → a4  ウ　b1 → a1 → a2 → a3 → a4 → b2 → b3 → b4  エ　b1 → b2 → b3 → a1 → a2 → a3 → a4 → b4  ア　それぞれ、３＋３、３×３が実行され、b4の結果が格納されます。最終値はｘ＝９となります。  イ　まず、タスクＡにてa1で３を保持し、その間にタスクＢで３×３が実行され、続けてタスクＡにて３＋９が実行され最終値は12となります。  ウ　まず、タスクＢにてb1で３を保持し、その間にタスクＡで３＋３が実行され、続けてタスクＢにて３×６が実行され最終値は18となります。  エ　アと同じ結果となります。  基本情報　平成17年度秋　問29　[出題頻度：★☆☆]  解答－イ |

#### ⑤排他制御

並行して動作している複数のプロセスが共有する資源に対して、データの変更を伴うアクセスを行う場合、データの整合性を保つために、データの読み書きを一時的に制限する必要が生じます。これを排他制御と呼びます。

プロセスの排他制御の代表的な方式に、**セマフォ**（semaphore）があります。セマフォには処理を要求するＰ操作（資源の獲得）と、処理の終了を通告するＶ操作（資源の解放）の２つの基本操作があり、一種のフラグとして使われるセマフォ変数によって排他制御を行います。

🏋プラスアルファ

**●排他制御の具体的な手順**

初めにセマフォ変数の初期値を１とし、

手順１：資源にアクセスするときＰ操作を実行する。このとき、

Ａ セマフォ変数が０でないならば、セマフォ変数から１を減じてプロセスの実行を続ける。

Ｂ セマフォ変数が０ならば、プロセスの実行を中断し、プロセスを待ち状態にする。

手順２：資源へのアクセス完了時、Ｖ操作を実行する。Ｖ操作はセマフォ変数に１を加えると共に、そのセマフォに対して待って  
　　　　　　 いるプロセスがあれば、Ｐ操作の再実行を行わせる。

|  |
| --- |
| 例題  セマフォを用いる目的として，適切なものはどれか。  ア　共有資源を管理する。  イ　スタックを容易に実現する。  ウ　スラッシングの発生を回避する。  エ　セグメンテーションを実現する。  セマフォとは、並行して動作しているプロセス間で同期を取ったり、割込み処理の制御を行う機構です。また、そのためにプロセス間で交換される信号を指すこともあります。特に、複数のプロセスでメモリ領域やディスク上のファイルを共有している場合に、同時にアクセスして内容の破壊や不整合が起きるのを防ぐために使われる技術です。  応用情報　平成30年度春　問18　[出題頻度：★☆☆]  解答－ア |

別冊演習ドリル 》 1-368,369

## 5. 入出力管理

学習のポイント

✅ スプーリング機能の仕組みと目的を覚えよう！

入出力管理の目的は、各種の入出力装置を、正確かつ効率よく動作させるように制御することです。

### １）入出力制御方式

代表的な入出力制御方式には、次に示すようなものがあります。

#### ①プログラム制御方式（直接制御方式）

CPUのレジスタを経由して、主記憶装置と入出力装置の間でデータ転送を行う方式です。CPUが直接に入出力装置を制御するため、CPUに無駄な待ち時間が生じ、CPUの使用効率が低下します。

#### ②DMA（Direct Memory Access）制御方式

CPUを介さずに、データ転送用のハードウェア（DMAコントローラ）によって、主記憶装置と入出力装置の間で直接転送を行う方式です。DMAコントローラは、入出力開始動作時にCPUから指示を受けてデータ転送を行い、データの転送完了は入出力割込みによってCPUへ伝えられます。なお、CPUと入出力装置が同時に主記憶装置にアクセスした場合には、入出力装置に優先権が与えられます。

データの流れ

制御の流れ

入出力指示

CPU

主記憶装置

DMAコントローラ

入出力装置

#### ③チャネル制御方式

主記憶装置と入出力装置の間に、入出力制御の機能を備えた専用の装置（チャネル）を置く方法です。チャネルは、チャネルプログラムの指令を解読しながら、主記憶装置と入出力装置の間のデータのやり取りを制御します。

データの流れ

制御の流れ

入出力指示

CPU

主記憶装置

チャネル

入出力装置

|  |
| --- |
| 例題  DMA制御方式による入出力処理の記述として，最も適切なものはどれか。  ア　CPUが入出力装置を直接制御することによって，データ転送が行われる。  イ　CPUを介さずに入出力装置と主記憶装置の間のデータ転送が行われる。  ウ　チャネル接続によって入出力装置と主記憶装置の間のデータ転送が行われる。  エ　入出力制御専用のプロセッサによってデータ転送を制御される。  ア　プログラム制御方式に関する記述です。  ウ　チャネル制御方式に関する記述です。  エ　チャネル制御方式に関する記述です。  ソフトウェア開発　平成19年度秋　問22　[出題頻度：★☆☆]  解答－イ |

### ２）スプーリング機能

スプーリング（スプール）機能は、磁気ディスクなどの補助記憶装置を仮想の入出力装置とすることによって、低速な入出力動作をプログラムの実行から切り離し、効率的なコンピュータの運用を図る機能です。具体的には、プリンタなどの低速な出力装置に比べて高速なハードディスクに一時的に全ての出力データを書き出し、CPUの待ち時間を減らし単位時間当たりの処理量（スループット）を向上させます。

|  |
| --- |
| 例題  スプーリングの説明として，適切なものはどれか。  ア　キーボードからの入力データを主記憶のキューに一旦保存しておく。  イ　システムに投入されたジョブの実行順序を，その特性や優先順位に応じて決定する。  ウ　通信データを直接通信相手に送らず，あらかじめ登録しておいた代理に送る。  エ　プリンタなどの低速な装置への出力データを一旦高速な磁気ディスクに格納しておき，その後に目的の装置に出力する。  スプーリングとは、低速な出力装置へデータを送るのではなく、一旦高速な磁気ディスクに格納することにより、CPUを効率よく利用する技術です。  ア　キーボードバッファに関する記述です。  イ　ジョブスケジューリングに関する記述です。  ウ　プロキシに関する記述です。  基本情報　平成27年度春　問17　[出題頻度：★★★]  解答－エ |

別冊演習ドリル 》 1-370～375

## 6. 記憶管理

学習のポイント

✅ 登場する用語は全て覚えよう！

✅ ページング方式は演習ドリルも含めて多くの問題演習をしておこう！

記憶管理の目的は、主記憶（メモリ）を効率よく使用するために、複数個のプログラムをどのように記憶領域に割り当てたり、解放したりするかを決めることです。

限られた主記憶を効率よく、より多くのプログラムに提供するために、実記憶方式から仮想記憶方式へと時代とともに次のように改善されてきました。

実記憶方式

単一区画方式

可変区画方式

固定区画方式

スワッピング方式

オーバレイ方式

ページング方式

セグメント方式

仮想記憶方式

### １）実記憶管理

マルチプログラミングという概念が存在していなかった初期のコンピュータでは１つのプログラムが記憶領域を占有して使い、使い終わった時点で次のプログラムに使用権を与える方法をとっていました。これを単一区画方式（単一連続割当て方式）と呼びます。その後、マルチプログラミングという概念が登場して、複数個のプログラムで主記憶を共有化する必要が生じたため、固定区画方式や可変区画方式などの割当て方式が考え出されました。

#### ①固定区画方式

固定区画（パーティション）方式は、主記憶（メモリ）を一定の区画に分割し、各区画にプログラムを読み込んで実行する方式です。この方式では、各区画の大きさが固定されているため、区画ごとに未使用領域が散在してしまいます。これを、**フラグメンテーション**（断片化または断片領域）といいます。

処理プログラム１

制御プログラム

処理プログラム２

処理プログラム３

フラグメンテーション（断片化）

＜主記憶＞

区画１

区画２

区画３

固定区画（パーティション）方式

#### ②可変区画方式

可変区画方式は、区画を廃止し、プログラムが必要とする未使用の領域を探して、これを割り当てる方式です。ただし、この方式でもフラグメンテーションの問題は解決しないため、次に示すようないくつかの対策が考えられました。

**ⅰ）コンパクション**

フラグメンテーションを解消するために、周期的に未使用領域を１つにまとめる動作をコンパクションと呼びます。コンパクションにより、各プログラムの主記憶上の領域を移動する再配置が行われます。なお、この再配置をプログラムの実行中に行うことを、**動的再配置**（dynamic relocation）といいます。また、プログラムが使用しなくなったメモリ領域を再び使用可能にする技術を**ガーベジコレクション**といい、通常ガーベジコレクションと同時にコンパクションが行われます。なお、ガーベジコレクションを行わないと、次第に利用可能なメモリが減っていく**メモリリーク**が起き、システムの性能が低下してしまうため、一定期間ごとに再起動を強いられることになります。

処理プログラム１

制御プログラム

処理プログラム２

処理プログラム３

区画１

区画２

区画３

処理プログラム３

処理プログラム１

制御プログラム

処理プログラム２

区画１

区画２

区画３

コンパクション

＜主記憶＞

＜主記憶＞

コンパクション

|  |
| --- |
| 例題  OSが記憶領域の割当てと解放を繰り返すことによって，細切れの未使用領域が多数できてしまう場合がある。この現象を何というか。  ア　コンパクション イ　スワッピング  ウ　フラグメンテーション エ　ページング  ア　コンパクションは、記憶装置上の細かく分割された空き領域をまとめることをいいます。  イ　スワッピングは、実記憶システムにおいて、主記憶装置と補助記憶装置の間で、必要に応じてプログラムを入れ替えることです。  エ　ページングは、仮想記憶システムにおいて、主記憶装置と補助記憶装置の間で、一定の大きさ（ページ単位）でプログラムを入れ替えることです。  基本情報　平成24年度秋　問18　[出題頻度：★★★]  解答－ウ |

別冊演習ドリル 》 1-376～380

**ⅱ）スワッピング**

スワッピングは、主記憶の使用効率を上げるために、必要がなくなったプログラムは補助記憶にロールアウト（スワップアウト）して、代わりに必要となるプログラムを主記憶にロールイン（スワップイン）します。

処理プログラム１

制御プログラム

処理プログラム２

＜主記憶＞

ロールアウト

＜補助記憶＞

処理プログラム１

処理プログラム２

ロールイン

スワッピング

**ⅲ）オーバレイ**

オーバレイは、プログラムを機能的なまとまりである**セグメント**と呼ばれる単位に分割し、実行時には必要なセグメントだけを補助記憶から読み込んで実行します。主記憶の容量よりも大きいプログラムを実行する場合に用いられます。

プログラムの構成 オーバレイ

共通セグメント

Ａ

(20ｋバイト)

Ｂ

(30kバイト)

Ｃ

(40kバイト)

プログラムの構成では90kバイトの領域が必要である。

Ａ

Ｂ

Ｃ

20k

30k

40k

60k

オーバレイにより、60kバイトの領域で実行が可能である。

なお、プログラムの分割、主記憶の割当て、および補助記憶から主記憶への読み込みは、全てプログラマの指示により行う必要があります。

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 例題  OSの記憶管理機能ａ～ｃに対応する適切な用語の組合せはどれか。   |  |  | | --- | --- | | 機能 | 特徴 | | ａ | あらかじめプログラムを幾つかの単位に分けて補助記憶に格納しておき，プログラムの指定に基づいて主記憶に読み込む。 | | ｂ | 主記憶とプログラムを固定長の単位に分割し，効率よく記憶管理する。これによって，少ない主記憶で大きなプログラムの実行を可能にする。 | | ｃ | プログラムを一時的に停止させ，使用中の主記憶の内容を補助記憶に退避する。再開時には，退避した内容を主記憶に再ロードし，元の状態に戻す。 |  |  |  |  |  | | --- | --- | --- | --- | |  | ａ | ｂ | ｃ | | ア | オーバレイ | ページング | スワッピング | | イ | スワッピング | オーバレイ | ページング | | ウ | スワッピング | ページング | オーバレイ | | エ | ページング | オーバレイ | スワッピング |   オーバレイやページングは、プログラムを分割して補助記憶に格納しておき、必要に応じて主記憶に読み込むことでプログラムを実行します。このとき、固定長でプログラムを分割するのは、ページング方式です。  ソフトウェア開発　平成20年度秋　問27　[出題頻度：★☆☆]  解答－ア |

別冊演習ドリル 》 1-381,382

### ２）仮想記憶方式

仮想記憶方式では、プログラムを補助記憶装置の１つである磁気ディスク（ハードディスク）装置に分割して格納しておき、実行に必要な部分だけを主記憶装置に読み込み、実行します。こうすることで見かけ上の主記憶（メモリ）の容量が増え、大きなプログラムも実行可能となりました。このとき、主記憶（メモリ）を実記憶、仮想メモリとして扱われる磁気ディスク装置の領域を仮想記憶と呼びます。

なお、プログラムは仮想記憶装置上に読み込まれるので、プログラムを実行するために**仮想アドレス**（仮想記憶装置上の論理アドレス）を**実アドレス**（実記憶装置上の物理アドレス）に変換する必要があります。その際使われる仮想アドレスと実アドレスの対応表を**アドレス変換テーブル**（**ページテーブル**）と呼びます。アドレス変換テーブルは、アクセス可能なアドレスの範囲を階層別に定めること（リング保護）で、記憶保護にも使われます。

アドレス変換の代表的な方法として、**動的アドレス変換機構**（**DAT**：Dynamic Address Translator）という、ハードウェアを用いて行う方法があります。

|  |
| --- |
| 例題  仮想記憶方式によって得られる効果はどれか。  ア　主記憶の実効アクセス速度が速くなる。  イ　主記憶の見かけの容量が拡大する。  ウ　消費電力が少なくなる。  エ　電源を切っても主記憶の情報が消えなくなる。  ア　磁気ディスクへのアクセスが増えるので、実効アクセス速度は遅くなります。  ウ　利用する装置が減るわけでないので、消費電力は減りません。  エ　主記憶の情報は、電源を切れば消えてなくなります。  初級システムアドミニストレータ　平成19年度秋　問7　[出題頻度：★☆☆]  解答－イ |

代表的な仮想記憶方式には、ページング方式があります。

#### ①ページング方式

ページング方式は、プログラムを一定の大きさであるページという単位に分割し、このページ単位で転送する方式です。そのため、一連のプログラムやデータを不連続な記憶領域に配置することができ、フラグメンテーションが発生しません。

仮想記憶装置から実記憶装置に必要なページを転送する動作を**ページイン**、逆に不要なページを実記憶装置から追い出す動作を**ページアウト**といいます。ページインやページアウトの動作は、**ページフォールト**と呼ばれる割込みによって行われます。

なお、ページング方式では、ページフォールトが多発し、処理効率が低下する場合があり、これを**スラッシング**といいます。ページフォールトの発生を抑制するには、メモリを増設するか、ページサイズを小さくするか、ジョブの多重度を下げてメモリの使用を抑制するか、のいずれかの方法をとる必要があります。

＜仮想記憶装置＞

ページ１

ページ４

ページ５

ページ６

ページ２

ページ３

ページ１

ページ４

制御プログラム

＜実記憶装置＞

ページイン

ページアウト

ページング方式

|  |
| --- |
| 例題  仮想記憶方式のコンピュータシステムにおいて、処理の多重度を増やしたところ，ページイン，ページアウトが多発して，システムの応答速度が急激に遅くなった。このような現象を何というか。  ア　オーバレイ イ　スラッシング  ウ　メモリコンパクション エ　ロールアウト  ア　オーバレイとは、プログラムを機能ごとにモジュールに分割し、実行時に必要なモジュールをロードすることです。  ウ　メモリコンパクションとは、記憶装置上の細かく分割された空き領域をまとめることです。  エ　ロールアウトとは、主記憶装置の内容を補助記憶装置に退避（スワップアウト）させることです。  基本情報　平成27年度春　問18　[出題頻度：★★★]  解答－イ |

別冊演習ドリル 》 1-383～385

ページング方式において、必要なページが実記憶装置にない場合は、ページフォールトが発生し、使用頻度の低いページを実記憶装置から追い出し、必要なページを実記憶装置に読み込みます。これをページリプレースメントといいます。

このとき、その時点以降に参照される頻度が最も低いページがどれかを推測し、実記憶装置のどのページを追い出すかを決定する方法をページングアルゴリズムといい、代表的な方法として次の二つがあります。

**・FIFO**（First In First Out）

FIFOは、最も古くページインされたページをページアウトする方法です。

**・LRU**（Least Recently Used）

LRUは、最後に参照されてからの経過時間が最も長いページをページアウトする方法です。

|  |
| --- |
| 例題  ページング方式の仮想記憶において，ページ置換えアルゴリズムにLRU方式を採用する。主記憶に割り当てられる  ページ枠が４のとき，ページ１，２，３，４，５，２，１，３，２，６の順にアクセスすると，ページ６をアクセスする時点で書き換えられるページはどれか。ここで，初期状態では主記憶にどのページも存在しないものとする。  ア　１ イ　２ ウ　４ エ　５  ページングの状況を図に示すと次のようになります。  ページアウト回数：４回  数字 ：ページインされたページ番号  数字：参照されたページ番号  １  ①  ②  ２  ③  ３  ④  ４  ２  ２  ①  １  ③  ３  ２  ２  ６  ⑥  ５  ⑤  ２  ３  ４  １  １  ３  ４  ５  ページアウトされたページ  基本情報　平成27年度春　問20　[出題頻度：★★★]  解答－エ |

別冊演習ドリル 》 1-386～389

# 2. ミドルウェア

## 1. ミドルウェアの役割と機能

学習のポイント

✅ 出題頻度を参考に効率よく学習しよう！

ミドルウェアは、基本ソフトウェアと応用ソフトウェアの中間に位置し、多数の応用ソフトウェアが共通に利用する基本処理機能を、標準化されたインタフェースで応用ソフトウェアから利用できるようにするためのソフトウェアです。

応用ソフトウェアで共通して利用される機能は、通常、OSの機能として実装されますが、これらの機能はどんな応用ソフトウェアでも、必ず必要とされるような基本的なものに限られます。そのため、特定の分野ではあるけれども、その分野では必ず必要とされるような機能や、多数の応用ソフトウェアで共通に利用する基本処理機能は、ミドルウェアで提供されることになります。データベース管理システム、通信管理システム、ソフトウェア開発支援ツール、運用管理ツールなどが、これに該当します。

|  |
| --- |
| 例題  ミドルウェアの説明として，適切なものはどれか。  ア　CPU，メモリや入出力装置などをアプリケーションソフトから利用するための基本的な機能を提供するソフトウェア  イ　基本ソフトウェアの機能を利用し，多様な利用分野に共通するより高いレベルの基本機能を提供するソフトウェア  ウ　経営資源の有効活用の視点から，経営資源を統合的に管理し，経営の効率化の実現を支援するソフトウェア  エ　電子メールや電子掲示板を介したコミュニケーション，データ共有，スケジュールの一元管理などの機能をもち，共同作業の支援を行うソフトウェア  ア　基本ソフトウェアに関する記述です。  ウ　ERP（Enterprise Resource Planning）パッケージに関する記述です。  エ　グループウェアに関する記述です。  初級システムアドミニストレータ　平成18年度秋　問28　[出題頻度：★☆☆]  解答－イ |

別冊演習ドリル 》 1-390

🏋プラスアルファ

**●シェルの役割と機能**

ジョブ管理、タスク管理、記憶管理など、主記憶装置上に常駐し、CPUに最も近いプログラム群をカーネル（kernel：核）と呼びます。利用者はこれらの機能を用いて、さまざまな処理を行います。しかし、カーネルは利用者と直接対話する能力を持っていないので、利用者との仲介役を果たすプログラムが必要となります。この仲介役のプログラムは、ユーザから見てカーネルの周りをすっぽり覆っている「殻」（shell）のように見えることからシェルと呼ばれています。

シェルは、ユーザが入力したコマンドを解釈してカーネルに処理を依頼し、その結果やメッセージなどを画面に表示する機能を持っています。なお、キーボードが標準入力、画面が標準出力になっていますが、リダイレクト機能を使用することで、標準入力、標準出力をファイルにすることもでき、標準出力にファイルを追加することもできます。

|  |
| --- |
| 例題　プラスアルファ  OSにおけるシェルの役割に関する記述として，適切なものはどれか。  ア　アプリケーションでメニューからコマンドを選択したり，設定画面で項目などを選択したりするといったマウス操作を，キーボードの操作で代行する。  イ　複数の利用者が共有資源を同時にアクセスする場合に，セキュリティ管理や排他制御を効率的に行う。  ウ　よく使用するファイルやディレクトリへの参照情報を保持し，利用者が実際のパスを知らなくても利用できるようにする。  エ　利用者が入力したコマンドを解釈し，対応する機能を実行するようにOSに指示する。  シェルとは，ユーザがキーボードから入力した文字やマウスのクリックなどを解釈して、対応した機能を実行するようにOSの中核部分（カーネル）に指示を伝えるソフトウェアのことです。  ア　ショートカットキーに関する記述です。  イ　DBMS（データベース管理システム）に関する記述です。  ウ　ショートカットに関する記述です。  初級システムアドミニストレータ　平成20年度秋　問8　[出題頻度：★☆☆]  解答－エ |

別冊演習ドリル 》 1-391

## ２. API（Application Programming Interface）

学習のポイント

✅ 例題を解くことができるように用語を覚えておこう！

アプリケーションから、OSが用意するウィンドウの描画やメニュー表示などの各種機能を利用するための仕組みを、**API**といいます。これらの機能は多くのアプリケーションに共通するもので、ソフトウェア開発においては、APIを利用することによって、その部分のプログラミングをする必要がなくなります。

なお、APIのモジュールの利用法には、動的リンクと静的リンクの２つがあります。**動的リンク**は必要なモジュールが何かをプログラムに記述しておき、プログラム実行時にOSが自動的にモジュールを呼び出し、プログラムに付加する方法です。**静的リンク**は、プログラムを作る際にあらかじめAPI機能を組み込んでしまう方法です。なお、動的リンクで呼び出すモジュールのことを**DLL**（Dynamic Link Library）と呼ぶこともあります。

|  |
| --- |
| 例題  OSにおけるAPI（Application Program Interface）の説明として，適切なものはどれか。  ア　アプリケーションがハードウェアを直接操作して，各種機能を実現するための仕組みである。  イ　アプリケーションから，OSが用意する各種機能を利用するための仕組みである。  ウ　複数のアプリケーション間でネットワークを介して通信する仕組みである。  エ　利用者の利便性を図るために，各アプリケーションのメニュー項目を統一する仕組みである。  APIは、OSがもつ、アプリケーション開発に利用できる命令や関数の集合体のことです。ウィンドウの描画やメニューの表示などアプリケーションの機能には共通しているものが多いため、各アプリケーションが共通して利用する機能をAPIが提供することにより、操作方法などを統一でき、開発時にも利用できるといった利点があります。  基本情報　平成20年度春　問29　[出題頻度：★★☆]  解答－イ |

別冊演習ドリル 》 1-392～394

# 3. ファイルシステム

## 1. ファイルシステム

学習のポイント

✅ 用語の暗記だけでなく、例題のような問題が解けるようにしよう！！

パソコンでは、数多くのファイルを効率よく管理するための仕組みとしてファイルシステムを用いています。

ファイルシステムは、ディレクトリとファイルから構成され、木構造（階層構造）で表されます。

／

DIR１

FILE２

DIR３

FILE４

FILE５

FILE３

DIR２

FILE１

ファイル

ファイル

ルートディレクトリ

サブディレクトリ

ファイルシステム（例）

**ディレクトリ**は、ファイルの情報を保存する登録簿であり、その下にファイルや別のディレクトリを置くことができます。ディレクトリは、階層構造をとることによって、効率的なファイル管理を可能にしています。ディレクトリには、ファイルと同様に名前を付けることが可能です。ただし、１つのディレクトリ内に、同じ名称のディレクトリやファイルを複数保存することはできません。階層構造の最上位にあるディレクトリを**ルートディレクトリ**、ルートディレクトリの配下にあるディレクトリをサブディレクトリと呼びます。

ディレクトリ内のファイルを指定する場合、ルートディレクトリから出発する方法と、現在操作を行っているディレクトリ（**カレントディレクトリ**）から出発する方法があります。前者を**絶対パス**による指定方法、後者を**相対パス**による指定方法といいます。パスとは通り道のことです。

これらの指定方法では、ディレクトリ名及びファイル名は経路順に“￥”で区切ります。

なお、“￥”の代わりに“／”を使用する表記方法を用いる場合もあります。

また、絶対パスは“￥”や“／”から始まり、“..”は１つ上のディレクトリを表します。

絶対パスの例　 〔ファイルシステム（例）〕中のFILE4

￥DIR1￥DIR3￥FILE4

相対パスの例　 〔ファイルシステム（例）〕中のDIR1をカレントディレクトリとしたときの、FILE4

DIR3￥FILE4

〔ファイルシステム（例）〕中のDIR3をカレントディレクトリとしたときの、FILE2

..￥FILE2

絶対パスと相対パス（例）

ファイルシステムを用いることで、アプリケーションプログラムは、ハードディスクやDVDなどの記憶媒体の違いに関係なく、統一したインタフェースでファイルにアクセス可能となります。

|  |
| --- |
| 例題  図の階層型ファイルシステムにおいて，カレントディレクトリが矢印で示すB1であるとき，ファイルC2を指定するものはどれか。ここで，ファイルの指定は，次の方法によるものとする。  A1  A2  /  B1  C1  B2  C2  〔指定方法〕  （1）ファイルは，“ディレクトリ名/…/ディレクトリ名/ファイル名”のように，経路上のディレクトリを順に“/”で区切って並べた後に“/”とファイル名を指定する。  （2）カレントディレクトリは“.”で表す。  （3）１階層上のディレクトリは“..”で表す。  （4）始まりが“/”のときは，左端にルートディレクトリが省略されているものとする。  （5）始まりが“/”，“.”，“..”のいずれでもないときは，左端にカレントディレクトリ配下であることを示す“./”が省略されているものとする。  また，図中の　　　　はディレクトリを表すものとする。  ア　../A1/B2/C2 イ　../B2/C2  ウ　A1/B2/C2 エ　B1/../B2/C2  解答群を確認すると、ルートディレクトリ（/）から指定する絶対パスによる記述がないため相対パスを利用することがわかります。相対パスとは、カレントディレクトリから指定のファイルにアクセスするための経路（パス）を指定する方法です。  この問題では、カレントディレクトリが“B1”のときのファイル“C2”の指定方法が問われています。したがって、まずは〔指定方法〕の(3)に記述されているとおり、１階層上のディレクトリに戻るために“..”を記述します。その後は、下位を指定していけばよいので、解答は「../B2/C2」となります。  初級システムアドミニストレータ　平成17年度春　問10　[出題頻度：★★★]  解答－イ |

別冊演習ドリル 》 1-395～399

## 2. ファイル編成とアクセス手法

学習のポイント

✅ 各ファイル編成方式の特徴を覚えよう！

### １）ファイルの構成要素

コンピュータで処理する１件分のデータを**レコード**と呼びます。また、同じ形式のレコードの集合体を**ファイル**と呼びます。レコードは通常、複数の項目（フィールド、アイテム）から構成されています。ファイル、レコード、フィールドの関係を図で示すと次のようになります。

ファイル

学生番号

氏名

国語得点

数学得点

クラス

レコード

フィールド

フィールド

フィールド

フィールド

フィールド

ファイル、レコード、フィールドの関係

|  |
| --- |
| 例題  ファイル，フィールド（項目），レコードの関係のうち，適切なものはどれか。ここで，“＞”の左側が上位の構成要素とする。  ア　ファイル＞フィールド＞レコード イ　ファイル＞レコード＞フィールド  ウ　フィールド＞ファイル＞レコード エ　フィールド＞レコード＞ファイル  学生番号や氏名、住所というように、意味を持った文字や数字の集まりをフィールド（項目）と呼び、これらのフィールドが集まって１件分のデータとなったときレコードと呼びます。ファイルは、一定の目的のもとに複数のレコードを集めたものです。  初級システムアドミニストレータ　平成19年度春　問10　[出題頻度：★☆☆]  解答－イ |

🏋プラスアルファ

**●ファイル編成方式**

ファイル編成方式は、アクセス手法と密接な関係があります。また、補助記憶媒体の種類によって、作成可能な編成が決まります。代表的な編成方式には次のようなものがあります。

**①順編成ファイル（sequential file）**

一連のレコードを物理的に連続した順序で記録したファイルです。

レコード1

レコード2

レコード3

レコードn

論理レコード

順編成ファイルの論理的構造

なお、順編成ファイルの特徴は次のとおりです。

・あらゆる記憶媒体に作成可能である。

・順次アクセス（物理的順序に従って読み書きする方式）だけが可能である。

・記憶媒体の利用効率が高い。

🏋プラスアルファ

**②直接編成ファイル（direct file）**

直接編成ファイルはハッシュ編成ファイルとも呼ばれ、レコードアドレスを用いて、特定のレコードだけを直接アクセスできるようにした  
　　ファイルです。レコードは、物理的な順序とは無関係に、指定したレコードアドレスに基づいて記録されます。

ファイルの利用者が特定のレコードを指定する場合、記憶媒体上での格納場所であるレコードアドレスの代わりに、キー値を指定  
　　します。このキー値を、プログラムでレコードアドレスに変換し、レコードの読み書きをします。

なお、直接編成ファイルの特徴は次のとおりです。

・磁気ディスクなどのDASD（Direct Access Storage Device：直接アクセス記憶装置）にのみ作成可能である。

・順次アクセス、直接アクセス（特定のレコードだけを直接読み書きする方式）、動的アクセス（直接アクセスと順次アクセスを  
　　　 組み合わせた方式）が可能だが、直接アクセス向きである。

・記憶媒体の利用効率が最も低い。

**③索引順編成ファイル（indexed file）**

索引順編成ファイルは、順編成ファイルに索引を付けたファイルです。索引は、特定レコードを直接アクセスする場合に用いられ  
　　ます。

索引順編成ファイルは、基本データ域、索引域、あふれ域から構成されます。

ａ．基本データ域

基本データ域は、ファイルを構成するレコードそのものを記録する領域で、キー順にレコードが記録されます。

ｂ．索引域

索引域は、キー項目を基にレコードの記録場所を表す索引を記録する領域です。

ｃ．あふれ域

あふれ域は、データの追加に伴い、基本データ域に格納できなくなったレコードを格納する領域です。

なお、索引順編成ファイルの特徴は次のとおりです。

・DASDにのみ作成可能である。

・レコードの追加や削除によって、あふれ域にデータが増えるとアクセス効率が低下し、処理に時間がかかるようになるためファイル  
　　　 の再編成が必要である。

・順次アクセス、直接アクセス、動的アクセスが可能である。

マスタ索引

シリンダ索引

キー項目

①

②

③

672

318

063

131

195

254

318

382

450

527

609

672

092

④

トラック索引

基本データ域

あふれ域

⑤

⑥

⑦

⑧

069

082

090

115

131

064

065

066

067

069

070

071

075

080

082

087

089

090

091

092

095

103

115

119

120

125

128

086

085

131

⑨

索引の検索順序

🏋プラスアルファ

**④区分編成ファイル（partitioned file）**

区分編成ファイルは、順編成ファイルをメンバと呼ばれる単位に分割し、メンバ単位でアクセスできるようにしたファイルであり、メン  
　　バの名称とアドレスなどが記録された登録簿（ディレクトリ）と呼ばれる記憶域が確保されています。特定のメンバをアクセスする  
　　場合、メンバを指定し、登録簿を検索します。

なお、区分編成ファイルの特徴は次のとおりです。

・DASDにのみ作成可能である。

・メンバ単位に直接アクセスが可能であり、メンバ内は順次アクセスを行う。

・代表的な使用例にプログラムライブラリがある。

登録簿

メンバ域

メンバＮ

メンバＥ

メンバＡ

メンバＧ

メンバＢ

アド

レス

Ａ

アド

レス

Ｂ

Ｅ

アド

レス

Ｇ

アド

レス

Ｈ

アド

レス

Ｎ

アド

レス

メンバＨ

区分編成ファイルのイメージ

|  |
| --- |
| 例題　プラスアルファ  ファイル編成に関する記述として，適切なものはどれか。  ア　区分編成ファイルは，ディレクトリとメンバから構成され，メンバ単位での更新はできない。  イ　索引編成ファイルは，直接アクセスと順アクセスの両方を可能としている。レコードの削除や挿入によって，アクセス効率や記録効率が低下することはない。  ウ　順編成ファイルは，レコードを順番に記録しているだけなので，キーによるアクセスはできないが，記録効率は高い。  エ　直接編成ファイルは，キーの値の分布にかかわらず，アクセス時間が一定であり，記録効率も高い。  ア　区分編成ファイルはメンバ単位での更新が可能です。  イ　索引編成ファイルはレコードの追加・削除が繰り返されると、基本データ域に使用できない領域（削除データが残っている部分）やあふれ域にデータが増えてしまい、アクセス効率や記録効率が低下します。  エ　直接編成ファイルは、キー値をもとに格納アドレスを求めるので、キー値が分散しているとアクセス時間や記録効率が低下します。  基本情報　平成13年度秋　問34　[出題頻度：★★☆]  解答－ウ |

別冊演習ドリル 》 1-400～403

## 3. バックアップ

学習のポイント

✅ バックアップの方法とそれぞれの特徴を覚えよう！

ハードウェアが故障したり、コンピュータウイルスに感染するなどの事故でデータが破損した場合に備えて、重要なデータはバックアップをとることが望まれます。

また、過去数回分のバックアップを保有（世代保存）することで、復旧させたい時点のデータに戻すことも可能になります。

バックアップは、データ更新のタイミングにより、夜間や休日を利用して定期的に行うのが一般的です。仮に業務時間中に行う場合には、バックアップ中のデータ更新を避けるため、ユーザからのアクセスを禁止してから行うべきです。

なお、バックアップに用いる媒体は将来のファイル容量を見積り、余裕ある媒体を用いる必要があります。さらにバックアップした媒体は、万が一、保管場所に災害が生じた場合に備えて、分散保管をしなければなりません。

|  |
| --- |
| 例題  サーバ上で稼働するアプリケーションプログラムが使用するデータのバックアップ処理に関する記述のうち，適切なものはどれか。  ア　アプリケーションプログラムの稼働状況と無関係に，毎日定刻に行う。  イ　サーバの運用管理者の都合の良い時刻に行う。  ウ　サービスレベルを低下させないように，アプリケーションプログラムを稼働したまま行う。  エ　バックアップ処理中のデータ更新を避けるため，アプリケーションプログラムを停止させてから行う。  バックアップ処理を行っている間に、アプリケーションプログラムがデータの内容を書き換えると、内容に不整合が生じてしまいます。したがって、バックアップ処理は、業務終了後や、業務中であれば当該データにアクセスできないようにして行う必要があります。  初級システムアドミニストレータ　平成13年度秋　問40　[出題頻度：★☆☆]  解答－エ |

別冊演習ドリル 》 1-404,405

バックアップの方法には、フルバックアップ（全バックアップ）と差分バックアップ、増分バックアップがあります。

**フルバックアップ**は、全てのデータをバックアップする方法です。増分バックアップや差分バックアップに比べてバックアップ時間はかかりますが、復元作業が簡単です。

**差分バックアップ**は、前回のフルバックアップからの変更部分だけをバックアップする方法です。毎回のバックアップ時間はフルバックアップに比べて短いものの、障害が発生した場合の復元には、フルバックアップをもとに差分を反映させるため、フルバックアップよりも時間がかかります。

**増分バックアップ**は、前回のバックアップからの変更部分だけをバックアップする方法です。毎回のバックアップ時間は３つの中で最も短いですが、障害が発生した場合の復元には、前回のフルバックアップをもとに順次増分を反映させるので、最も時間がかかります。

〔バックアップ方法の比較〕

データ：毎日、新しいデータが追加される。

Ａ

Ａ

Ｃ

Ｄ

Ｅ

Ａ

Ｂ

Ｃ

Ｄ

Ａ

Ｂ

Ｃ

Ａ

Ｂ

1日目

2日目

3日目

4日目

5日目

Ｂ

・フルバックアップ：毎日全てのデータをバックアップする。

Ａ

Ｅ

5日目

Ｃ

Ｄ

Ｂ

Ａ

Ｃ

3日目

Ｂ

Ａ

フルバックアップ

Ａ

Ｂ

2日目

Ａ

Ｂ

Ｃ

Ｄ

4日目

Ａ

Ｅ

5日目

Ｄ

Ｂ

Ｃ

Ａ

Ｃ

3日目

Ａ

フルバックアップ

Ａ

Ｂ

2日目

Ｂ

Ａ

Ｃ

Ｄ

4日目

Ｂ

Ａ

Ｅ

5日目

Ｂ

Ｃ

Ｄ

Ａ

Ｃ

3日目

Ｂ

Ａ

フルバックアップ

Ａ

Ｂ

2日目

Ａ

Ｂ

Ｃ

Ｄ

4日目

・差分バックアップ：フルバックアップ後、追加された部分（差分）を毎日バックアップする。

・増分バックアップ：フルバックアップ後、その日に追加された部分を毎日バックアップする。

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 例題  Ａ社は業務で使用しているサーバのデータをサーバのハードウェア障害に備えてバックアップをしたいと考えている。次のバックアップ要件を満たす計画のうち，Ａ社のバックアップ計画として適切なものはどれか。  〔バックアップ要件〕  ・サーバ障害時には障害が発生した前日の業務終了後の状態に復旧したい。  ・業務で日々更新するデータは全体に比ベてごく少量だが，保有しているデータ量が多く，フルバックアップには時間が掛かるので，月曜日～土曜日にはフルバックアップを取ることができない。   |  |  |  | | --- | --- | --- | |  | バックアップ方法 | バックアップファイル保存場所 | | ア | 月曜日～土曜日にはバックアップを取得せず，日曜日にフルバックアップを取得する。 | 外部メディアへ出力して所定の場所で，それを保管する。 | | イ | 月曜日～土曜日にはバックアップを取得せず，日曜日にフルバックアップを取得する。 | 障害時にすばやく復旧させるためにサーバ内部のフォルダへ置く。 | | ウ | 日曜日にフルバックアップを取得し，月曜日～土曜日には，フルバックアップ以降に更新や追加，削除された部分のデータを差分バックアップとして取得する。 | 外部メディアへ出力して所定の場所で，それを保管する。 | | エ | 日曜日にフルバックアップを取得し，月曜日～土曜日には，フルバックアップ以降に更新や追加，削除された部分のデータを差分バックアップとして取得する。 | 障害時にすばやく復旧させるためにサーバ内部のフォルダへ置く。 |   ア、イ　日曜日に取得したバックアップしか存在しないので、障害が発生した日の直前の日曜日の状態にしか復旧できません。  イ、エ　同一サーバ内にバックアップファイルを保存しているので障害を復旧することはできません。  ITパスポート　平成24年度春　問81　[出題頻度：★★★]  解答－ウ |

別冊演習ドリル 》 1-406～410

# 4. 開発ツール

ソフトウェアの生産性や品質を高めるために、各種の開発支援ツールが利用されています。

## 1. 開発支援ツールの種類と特徴

学習のポイント

✅ 上流CASEツールと下流CASEツールの支援範囲を明確にしておこう！

開発支援ツールには次のようなものがあります。

内部設計の各作業を

支援するためのツール

プログラムの作成及び入力・実行を支援するためのツール

プログラムのテストを

効率よく進めるためのツール

ソフトウェア開発の各工程に提供されるツール

開発支援ツール

内部設計支援ツール

プログラミング支援ツール

テスト支援ツール

CASEツール

・・・

・・・

・・・

・・・

開発支援ツールの種類

### １）CASE（Computer Aided Software Engineering）ツール

CASEツールは、ソフトウェア開発の各工程に提供される開発支援ツールであり、これによって生産性、品質、保守性などの向上が期待できます。

統合CASEツール

基本計画

外部設計

内部設計

プログラム設計

プログラミング

テスト

運用・保守

共通ＣＡＳＥツール

（共通ユーザインタフェース）

上流CASEツール

下流CASEツール

保守CASEツール

開発プラットフォームサービス

提供ＣＡＳＥツール

開発

リポジトリ

ライブラリ

CASEツール

CASEツールは、支援範囲の違いなどにより、次のように分類できます。

#### ①上流CASEツール

基本計画、外部設計、内部設計といったシステム開発の上流工程における作業を支援するツールです。

#### ②下流CASEツール

プログラミング、テストといったシステム開発の下流工程における作業を支援するツールです。

#### ③保守CASEツール

保守CASEツールは、保守工程における作業を支援するツールです。

#### ④共通CASEツール

共通CASEツールは、プロジェクト管理、文書の作成支援といった開発工程全体の作業を支援するツールです。

#### ⑤開発プラットフォームサービス提供CASEツール

開発プラットフォームサービス提供CASEツールは、ツールの統合化を進める際に使われるツールで、CASEツール間のインタフェースを定義する役割を果たすものです。

#### ⑥統合CASEツール

統合CASEツールは、システムのライフサイクルの全工程をカバーするツールです。

|  |
| --- |
| 例題  CASEツールが提供する機能のうち，上流CASEツールに属するものはどれか。  ア　DFDの作成支援 イ　テストデータの作成支援  ウ　プログラムのコードの自動生成 エ　ライブラリの管理支援  CASEツールが提供する機能のうち、上流CASEツールは、開発工程における基本計画・外部設計・内部設計・プログラム設計に対する支援ツールです。選択肢のうち、DFD（Data flow Diagram）は基本計画にて、業務システムをモデル化するのに用いるため、DFDの作成支援は上流CASEツールとなります。  基本情報　平成21年度秋　問20　[出題頻度：★★☆]  解答－ア |

別冊演習ドリル 》 1-411

## 2. 言語処理ツールの種類と特徴

学習のポイント

✅ 翻訳プログラムの種類と特徴を覚えよう！

✅ コンパイラの手順を覚えよう！

機械語以外のプログラム言語で記述されたプログラムは、そのままコンピュータで実行することはできません。そのため言語プロセッサ（言語処理プログラム）を用いてコンピュータが理解できる状態に翻訳してから利用します。言語プロセッサは、翻訳処理の違いにより、インタプリタ（解釈プログラム）と翻訳プログラムに分類されます。

言語プロセッサ

アセンブラ

コンパイラ

ジェネレータ

翻訳プログラム

インタプリタ

言語プロセッサの種類

### １）インタプリタ

インタプリタ（解釈プログラム）は、命令を一文ずつ解釈しながら実行する言語プロセッサで、目的プログラム（オブジェクトプログラム）を作成しません。そのため、プログラムが未完成でも完成部分から実行することができますが、実行時に機械語に翻訳しながら動作するので、コンパイラなどの翻訳プログラムに比べて処理速度が遅くなります。インタプリタを用いる言語には、BASICやPerl（パール）、JavaScript、Python（パイソン）などがあります。

|  |
| --- |
| 例題  インタプリタ方式の言語処理に関する記述のうち，適切なものはどれか。  ア　コンパイラ方式に比べ，プログラムの実行速度は遅い。  イ　再帰的な関数呼出しは許されない。  ウ　ソースプログラムをあらかじめ機械語に翻訳して実行する。  エ　対話的な実行環境の構築には不向きである。  インタプリタは、目的プログラムを生成しません。実行時に機械語に翻訳しながら動作するので、コンパイラなどの翻訳プログラムに比べて処理速度が遅くなります。  初級システムアドミニストレータ　平成17年度秋　問22　[出題頻度：★☆☆]  解答－ア |

別冊演習ドリル 》 1-412,413

### ２）翻訳プログラム

翻訳プログラムには、アセンブラ、コンパイラ、ジェネレータがあります。

**アセンブラ**は、アセンブラ言語で書かれた原始プログラム（ソースプログラム）をコンピュータが理解できる機械語の目的プログラムに翻訳します。

**コンパイラ**は、CやCOBOL、Java、Pythonなどの高水準言語で書かれた原始プログラムを目的プログラムに翻訳するものです（PerlやPythonなど、現在でも利用されているプログラム言語の多くは、インタプリタとコンパイラ両方の実行形式に対応しています）。なお、疑似命令（付加的に定義された機能）を利用して作成された原始プログラムをコンパイラが翻訳可能な命令に書き換えるプログラムを**プリコンパイラ**と呼びます。

**ジェネレータ**は、入力・処理・出力に関する条件をパラメタとして与えることで自動的にプログラムを生成します。そのため生成プログラムとも呼ばれます。

また、高水準言語で作成した原始プログラムを翻訳し、開発に使用しているものとは異なる環境で実行できる機械語のプログラムを生成する翻訳プログラムを、**クロスコンパイラ**と呼びます。

擬似命令言語

目的プログラム

原始プログラム

実行

目的プログラム

原始プログラム

実行

コンパイラ

アセンブラ

クロスコンパイラ

クロスアセンブラ

シミュレータ

エミュレータ

トランスレータ

プリプロセッサ

〔Ｂ社システム〕

〔Ａ社システム〕

|  |
| --- |
| 例題  あるコンピュータ上で，異なる命令形式のコンピュータで実行できる目的プログラムを生成する言語処理プログラムはどれか。  ア　エミュレータ イ　クロスコンパイラ  ウ　最適化コンパイラ エ　プログラムジェネレータ  高水準言語で作成したソースコードを解釈し、開発に使用しているものとは異なる環境で実行可能な機械語のプログラムを生成するソフトウェアをクロスコンパイラといいます。  ア　エミュレータは、あるシステム上でほかのOSやCPUの機能を再現し、そのOS／CPU向けのアプリケーションソフトを動作させるためのものです。  ウ　最適化コンパイラは、プログラムの動作環境で、処理を最も速く実行できる形式に変換することを目的としたコンパイラです。  エ　プログラムジェネレータは、入力・処理・出力に関する条件をパラメタとして与えることにより、自動的にプログラムを生成する言語プロセッサです。  応用情報　平成27年度春　問19　[出題頻度：★★☆]  解答－イ |

別冊演習ドリル 》 1-414～416

### ３）コンパイラの処理手順

コンパイラの処理手順は次の５段階に分けることができます。

#### ①字句解析

原始プログラムの文字列から、字句（トークン）を切り出します。字句は構文を構成する名前や定数などの最小単位です。有限オートマトンにより切り出しを行います。

#### ②構文解析

数式などが構文規則に従っているかどうかを検査します。構文解析の技法として、構文木や逆ポーランド記法が使用されます。

#### ③意味解析

構文解析で決まった字句の属性において、矛盾が生じていないかを検査します。例えば、算術式に文字型の変数が使用されていないかなどを検査します。この段階で内部的なコードである中間コードの生成を行います。

#### ④最適化

プログラムの実行時間や容量が少なくなるように、式の簡略化やループの再編など、プログラムの変更をします。

#### ⑤コード生成

中間コードから目的プログラムを生成します。

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 例題  図はコンパイラにおける処理の流れを表している。ａ～ｄに入る用語の組合せとして，適切なものはどれか。  ａ  ｂ  ｃ  ｄ  目的コードの生成   |  |  |  |  |  | | --- | --- | --- | --- | --- | |  | ａ | ｂ | ｃ | ｄ | | ア | 構文解析 | 字句解析 | 意味解析 | 最適化 | | イ | 構文解析 | 字句解析 | 最適化 | 意味解析 | | ウ | 字句解析 | 構文解析 | 意味解析 | 最適化 | | エ | 字句解析 | 構文解析 | 最適化 | 意味解析 |   まず、与えられた原始プログラム（ソースプログラム）に記述されている内容から、名前、数値、定数、識別子などを取り出す処理である字句解析を行い、次に字句の構成を調べる構文解析、そして字句の属性が相互に正しい関係になっているかどうかを調べる意味解析、目的のプログラムのサイズや実行時間が小さくなるようにプログラムを変更する最適化処理、最後に目的プログラムを作るためのコードを生成します。  ソフトウェア開発　平成13年度春　問44　[出題頻度：★★☆]  解答－ウ |

別冊演習ドリル 》 1-417～420

🏋プラスアルファ

**●プログラムの実行手順**

目的プログラムは、**リンカ**（リンケージエディタ、連係編集プログラム）によってライブラリのプログラムが複数組み合わされて実行可能プログラム（ロードモジュール）に変換され、補助記憶装置に格納されます。その後、**ローダ**によって取り出されて、主記憶装置の指定された番地に格納され、実行されます。

コンパイラ

リンカ

ローダ

プログラム

ライブラリ

原始プログラム

目的プログラム

実行可能プログラム

実行

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 例題　プラスアルファ  図はプログラムを翻訳して実行するまでの流れを示したものである。コンパイラ，リンカ，ローダの入出力の組合せとして，適切なものはどれか。  原始プログラム  コンパイラ  リンカ  ローダ  ａ  ｂ  ｃ   |  |  |  |  | | --- | --- | --- | --- | |  | ａ | ｂ | ｃ | | ア | 目的プログラム | ライブラリモジュール | ロードモジュール | | イ | ライブラリモジュール | ロードモジュール | 目的プログラム | | ウ | ロードモジュール | 目的プログラム | ライブラリモジュール | | エ | ロードモジュール | ライブラリモジュール | 目的プログラム |   リンカは、ライブラリの複数のプログラムを組み合わせて実行可能プログラム（ロードモジュール）に変換するプログラムです。  したがって、空欄ａには、目的プログラムが入り、空欄ｃには、ロードモジュールが入ることがわかります。  基本情報　平成21年度春　問22　[出題頻度：★☆☆]  解答－ア |

別冊演習ドリル 》 1-421

# 5. オープンソースソフトウェア

## 1. オープンソースソフトウェア（OSS：Open Source Software）

学習のポイント

✅ 例題や演習ドリルの問題を解きながら、知識を身に付けていこう！

オープンソースソフトウェアは、ソースコードが公開されているソフトウェア群のことで、誰でも改良や再配布が可能です。代表的なオープンソースソフトウェアに、Linux（OS）、BIND（DNSサーバ）、Postfix（メールサーバ）、Apache（Webサーバ）、PostgreSQL（DBMS）、Eclipse（統合開発環境）、Thunderbird（メールソフト）、Firefox（ブラウザ）などがあります。

また、Linuxなどのように、改変したものを販売している業者（ディストリビュータ）もあります。ただし、改良や再配布に際しては原著者の著作権表示を義務付けています。

OSSの概念をまとめたものに、OSI（Open Source Initiative）が公表したOSD（The Open Source Definition）があり、次のような基準があります。

①自由な再配布ができる。

②ソースコードを入手できる。

③派生物が存在でき、派生物に同じライセンスを適用できる。

④差分情報の配布を認める場合には、同一性の保持を要求してもかまわない。

⑤個人やグループを差別しない。

⑥適用領域に基づいた差別をしない。

⑦再配布において追加ライセンスを必要としない。

⑧特定製品に依存しない。

⑨同じ媒体で配布される他のソフトウェアを制限しない。

⑩技術的な中立を保っている。

OSSに関するライセンスとしては、FSF（Free Software Foundation）が提唱したGPL（General Public License）や、カリフォルニア大学バークレー校（University of California, Berkeley, UCB）が開発・配布したBSD（Berkeley Software Distribution）ライセンスがあります。

|  |
| --- |
| 例題  Linuxに代表されるソフトウェアであって，再配布の自由，再配布時のソースコード包含，派生ソフトウェア改変の許諾などが要求されるものを何というか。  ア　オープンソースソフトウェア イ　コンポーネントウェア  ウ　シェアウェア エ　ミドルウェア  イ　コンポーネントウェアとは、オブジェクト指向技術を基盤としたソフトウェア部品を組み立てることによって，アプリケーションを開発する技術の総称のことです。  ウ　シェアウェアとは、無償で試用することができるが、試用期間後も引き続き使用する場合には、使用料を支払わなければならないソフトウェアのことです。  エ　ミドルウェアとは、基本ソフトウェアの機能を利用し、多様な利用分野で共通に使用する基本処理機能を提供するソフトウェアのことです。  基本情報　平成20年度春　問40　[出題頻度：★★★]  解答－ア |

別冊演習ドリル 》 1-422～425

🏋プラスアルファ

**●UNIX**

UNIXは、複数のユーザが同時に利用することが可能なマルチユーザ機能や、複数のプログラムを同時に実行することが可能なマルチタスク機能を備えたオペレーティングシステムです。中核をなすカーネルがＣ言語で記述されているため、異なるプラットフォームに対する移植性が高いソフトウェアです。なお、UNIXなどのマルチタスクOSにおいて、OSに常駐して、バックグラウンドで動作し、さまざまなサービスを提供する機能をデーモンと呼びます。

UNIXではファイルを、テキスト、オブジェクトコード、画像データなどを格納するための通常ファイル、ファイル名とファイルの実体を対応付けるためのディレクトリファイル及び磁気ディスクなどの入出力装置にアクセスするための特殊ファイルの３種類に分類しています。

そしてUNIXでは、全てのディレクトリとファイルを単一のディレクトリツリーで管理しています。ファイルの格納先が複数の媒体に分散していても、マウントという操作でディレクトリツリーのどこかに接ぎ木され、管理されます。また、ネットワーク経由でファイルをマウントできるNFS（Network File System）を用いることで、UNIXマシン間でマウント操作を行えば、ネットワーク経由のリモートファイルもディレクトリに割り付けてアクセスすることができます。

またUNIXでは、自分のログイン名のついた自分専用のディレクトリがあり、これをホームディレクトリと呼びます。各ユーザは、ファイルの保存などに使用しており、パーミッション（ファイルのモード）を指定することができます。通常は、ほかのユーザに見てもらうために、リードパーミッション（参照モード）にしています。

UNIXにおいて複数のコマンドでデータを連続的に処理するときに、コマンド間でデータを受け渡す仕組みをパイプと呼びます。パイプは、あるプログラムの出力を、ファイルなどに出力することなく、別のプログラムに入力することができる機能です。パイプは連続して使用できるので、あるプログラムの出力を別のプログラムに入力し、さらにその出力を別のプログラムに入力する、といった具合に、一時ファイルを作成することなく、複数の処理を一度に実行することができます。

|  |
| --- |
| 例題　プラスアルファ  UNIXの階層的ファイルシステムにおいて，アカウントをもつ一般の利用者がファイル保存などに使う階層で最上位のものはどれか。  ア　カレントディレクトリ イ　デスクトップディレクトリ  ウ　ホームディレクトリ エ　ルートディレクトリ  UNIXでは、自分のログイン名のついた自分専用のディレクトリがあり、これをホームディレクトリと呼びます。各ユーザは、ファイルの保存などに使用しており、パーミッション（ファイルのモード）を指定することができます。通常は、ほかのユーザに見てもらうために、リードパーミッション（参照モード）にしています。  基本情報　平成22年度春　問21　[出題頻度：★☆☆]  解答－ウ |

別冊演習ドリル 》 1-426