

### ③オペレーティングシステムの構造

基礎OS 2015年度(1組)

### 問1 割り込みとOS

多重プログラミングは、I/O処理中に他のプログラムを実行することによりCPUの使用効率を高める技術である。これを実現するために、イベントの発生によりOSを起動する機構は、以下のどれか。

- A. オブジェクト指向プログラミング
- B. 0番地ジャンプ
- C. システムコール
- D. システムプログラミング
- ☒ E. 割り込み
- F. IOCS

多重プログラミングはCPUの使用効率を高めることを目的としている。入出力が始まると他のプログラムにCPUを切替え、実行中状態にする。また、入出力が終わったプログラムは、実行中にすることができる。これを実現するためには、入出力の開始、終了を検出する必要があり、その機構が割り込みである。

### 問2 割り込みの種類

外部割り込みに分類されるのはどれか。(基本情報 平成17年度秋期 問20改)

- ☒ A. I/O装置の動作が終わったときに発生する割り込み
- B. I/O要求など、OSに対してサービスを依頼したときに発生する割り込み
- C. 定義されていない命令を実行しようとしたときに発生する割り込み
- D. 割り算命令で、ある数をゼロで割ろうとしたときに発生する割り込み

次の次のスライド参照

### 問3 割り込みの種類

以下で内部割り込みはどれか。

- A. ネットワークインタフェースカードが故障した場合に発生する割り込み
- B. アプリケーションプログラムがレジスタの内容をメモリに格納する際のエラーにより発生した割り込み
- C. アプリケーションプログラムが要求したディスクアクセスが完了したときに発生する割り込み
- ☒ D. アプリケーションプログラムがOSにより提供される機能を要求した場合に発生する割り込み
- E. ディスクへの書き込みが完了したときに発生する割り込み

次のスライド参照

### 重要: 割り込みとOSの起動

- ハードウェアがイベント(割り込み原因)を検出。割り込みによりOS(カーネル)を起動
  - そのとき実行していたプログラムを中断し、OS(カーネル)の特定番地にジャンプ
  - OS(カーネル)は、割り込みによってのみ起動される

割り込みの種類(注1)と割り込み原因

- 外部割り込み(external interrupt): ハードウェア割り込みとも呼ばれる
  - ハードウェアが原因で発生する割り込み
    - マシンチェック割り込み: ハードウェアの故障、メモリの読み出しエラーなど
    - 入出力割り込み: I/O完了(I/O装置の動作完了通知)、異常通知など
    - 時計(タイマ)割り込み: 設定された時間の超過
- 内部割り込み(internal interrupt): ソフトウェア割り込みとも呼ばれる
  - プログラムの実行が原因で発生する割り込み
    - プログラムの誤り: 演算例外(注2)、不正命令(注3)、メモリ保護違反(注4)
    - システムコール: アプリケーションプログラムによるシステムコールの発行

- 注1: 外部割り込みを「割り込み」、内部割り込みを「割り出し(trap)」と言う人もいる
- 注2: ゼロ除算、オーバフロー、注3: 定義されていない命令コードの使用
- 注4: 許可されていないメモリ領域へのアクセス

### 問4 割り込み処理

プログラムの実行中に割り込みが発生すると、ハードウェア処理により、OS(カーネル)にジャンプする。カーネルは、割り込まれたプログラムの再開に備えて、以下のどれを行うか。

- ☒ A. レジスタ退避
- B. メモリ管理
- C. スケジューラ起動
- D. ファイル管理
- E. 同期制御
- F. 割り込み命令

次の次のスライド参照

## 問5 割り込み処理

OS2015③-1

割り込み処理終了後に、割り込まれたプログラムを再開する際に必要な情報で、割り込みが発生した時に実行していた命令の次の命令のアドレスは、以下のどれに格納されているか。

- ☒ A. プログラムカウンタ
- ☐ B. ベースレジスタ
- ☐ C. 命令フェッチレジスタ
- ☐ D. 汎用レジスタ
- ☐ E. メモリ管理レジスタ
- ☐ F. スーパーバイザ

次のスライド参照

割り込みベクトル、割り込み分析、レジスタ退避、プログラムカウンタの意味を理解すること

## 重要: 割り込み処理

OS2015③-1

- ・ ハードウェアの処理
  - ①プログラムの実行中に割り込みの事象が発生(これを検出)
  - ②**割り込みベクトル**(割り込み原因番号)をメモリの指定番地に設定
  - ③OS(カーネル)の指定番地(割り込み処理ルーチンの入り口)にジャンプ
- ・ OS(カーネル)の処理
  - ④**レジスタ退避**(レジスタの内容(注1))をプログラム実行の管理エリア(注2)に保存)
  - ⑤**割り込み分析**により、原因ごとの割り込み処理を選択(注3)
  - ⑥選択した割り込み処理にジャンプし、実行を開始

・ 注1: **プログラムカウンタ**(割り込まれた命令の次に実行するはずだった命令の番地)および各種汎用レジスタの内容。割り込まれたプログラムを再開する際、元の値に戻す必要がある。

・ 注2: 実行中のプログラムは、**プロセス**として管理されている。この管理エリアは各プロセスに割り当てられ、**PCB**(Process Control Block)という。

・ 注3: 割り込みベクトルの値をインデックスとした配列に割り込み処理プログラムの番地を記述したEDT(割り込みベクトルテーブル)からジャンプ先番地を取り出す。

## 問6 I/O制御

OS2015③-1

I/O装置がメモリを直接参照することで、データの転送処理からCPUを解放し、CPUの負荷を軽減する方式を何と言うか。

- ☐ A. CDE
- ☐ B. IDL
- ☐ C. IPC
- ☐ D. TSS
- ☒ E. DMA

DMAの処理

CPUは、I/O装置にデータの場所と指示を送る

I/O装置は、I/O制御装置を介して、メモリを直接参照

(CPUはデータの転送から解放され、プログラムの実行に専念できる)

I/O完了は、割り込みによって通知される

## 問7 I/O処理の効率化

OS2015③-1

アプリケーションプログラムのI/O要求で受け渡される論理的なデータの単位を何と言うか。

尚、ブロッキングは、このデータの単位の複数個を1つのブロックにまとめて入出力し、処理を効率化する方法である。

- ☐ A. ストリーム
- ☒ B. レコード
- ☐ C. ブロック
- ☐ D. バッファ
- ☐ E. キャッシュ

アプリケーションプログラムは**レコード**単位に処理を行うので、**レコード**単位にI/O要求をしてくる。

アプリケーションプログラムが要求する度に、1レコードずつ入出力するよりも複数のレコードをまとめて**ブロック**とし、1回で入出力を行った方が効率が良い。

(I/O処理の回数を減らせる)

この方法を**ブロッキング**と呼ぶ。

ブロックはOSがI/O装置にアクセスする単位で  
ブロック長>レコード長

## 問8 ブロッキングの効果1

OS2015③-1

1レコード800Bのデータを5000件出力する。1ブロック4096Bでブロッキングする場合、ブロッキングしない場合と比べ、ディスクのアクセス回数が何%削減されるか。【必要なら%値の小数点以下を四捨五入し、整数(整数部)のみを半角数字で記入。(記入例 73)】

答 80 [%]

ブロック化係数

$4096 / 800 = 5.12 \rightarrow 5$

ブロック化しない場合のアクセス回数=5000

ブロック化した場合のアクセス回数

$5000 / 5 = 1000$

削減されたアクセス回数

$5000 - 1000 = 4000$

削減率

$4000 / 5000 = 0.8 = 80\%$

または、ブロック化したアクセス回数は、元のアクセス回数の何%かを計算

$1000 / 5000 = 0.2 = 20\%$

削減率

$100\% - 20\% = 80\%$

## 問9 ブロッキングの効果2

OS2015③-1

1レコード2000Bのデータを5000件出力する。1ブロック4096Bでブロッキングする場合、ブロッキングしない場合と比べ、ディスクのアクセス回数が何%削減されるか。【必要なら%値の小数点以下を四捨五入し、整数(整数部)のみを半角数字で記入。(記入例 73)】

答 50 [%]

ブロック化係数

$4096 / 2000 = 2.048 \rightarrow 2$

ブロック化しない場合のアクセス回数=5000

ブロック化した場合のアクセス回数

$5000 / 2 = 2500$

削減されたアクセス回数

$5000 - 2500 = 2500$

削減率

$2500 / 5000 = 0.5 = 50\%$

または、ブロック化したアクセス回数は、元のアクセス回数の何%かを計算

$2500 / 5000 = 0.5 = 50\%$

削減率

$100\% - 50\% = 50\%$

問10 ブロッキングの効果3

1レコード200Bのデータを50件出力する。1ブロック4096Bでブロッキングする場合、ブロッキングしない場合と比べ、ディスクのアクセス回数が何%削減されるか。【必要なら%記号の小数点以下を四捨五入し、数値(整数部)のみを半角数字で記入。(記入例 73)】

答 94 (94%)

ブロック化係数

$4096 \div 200 = 20.48 \rightarrow 20$

ブロック化しない場合のアクセス回数=50

ブロック化した場合のアクセス回数

$50 \div 20 = 2.5 \rightarrow 3$

削減されたアクセス回数

$50 - 3 = 47$

削減率

$47 \div 50 = 0.94 = 94\%$

または、ブロック化したアクセス回数は、元のアクセス回数の何%かを計算

$3 \div 50 = 0.06 = 6\%$

削減率

$100\% - 6\% = 94\%$