* 1. Prozess

Ein Prozess ist ...

Der Code und Zustand eines Programmes

1.2

Welche(s) Betriebsmittel wird/werden einem Prozess in Ausführung in jedem Fall zugewiesen?

Prozessor

Hauptspeicher

1.3

Das Konzept der virtuellen Prozessoren ermöglicht es, ...

Prozesse verzahnt auszuführen.

2.1 Prozesskontrollblock, PCB

Wozu dient der Prozesskontrollblock (PCB)?

Zugriff auf Programminformationen.

2.2

Welche Aussage(n) ist/sind wahr für Prozesse in statischen Betriebssystemen?

Zur Laufzeit können keine neuen Prozesse erzeugt werden.

Der Speicherort für Programmcode und Programmzustand kann sich nicht ändern während der Laufzeit.

Der Prozesskontrollblock ist Teil des Programmzustandes.

2.3

Welche Aussage(n) ist/sind wahr für Prozesse in dynamischen Betriebssystemen?

Die Anzahl der Prozesse kann sich während der Laufzeit verändern.

Prozesse können andere Prozesse erzeugen.

3.1 Dispatching

Was bezeichnet man im Prozess-Kontext als Dispatching?

Unterbrechung der Abarbeitung eines Programmes und Fortführung eines anderen Programmes

3.2

Welche Konsequenzen hat das Umschalten zwischen zwei Prozessen in jedem Fall?

Änderung des Befehlszeigers

Unterbrechung der Ausführung eines Programmes durch den Prozessor

3.3

Welche Aussage(n) ist/sind wahr für das Umschalten zwischen Prozessen durch Sprünge?

Ein Prozess "entscheidet" selbst, wann die Umschaltung zu einem anderen Prozess erfolgt.

Das Betriebssystem hat keinen Einfluss auf den Zeitpunkt der Umschaltung zwischen Prozessen.

3.4

Warum ermöglicht das Merken der Fortsetzstelle bei der Umschaltung zwischen zwei Prozessesn eine größere Flexibilität?

Der laufende Prozess kann an einer "beliebigen" Stelle unterbrochen werden und die Ausführung wird später an der entsprechenden Stelle fortgesetzt.

3.5

Was bezeichnet der Begriff "Scheduling"?

Festlegung einer Ausführungsreihenfolge von Prozessen

3.6

Was umfasst der Mikrozustand eines Prozesses?

alle Informationen, die sich während der Ausführung eines Prozesses in den Registern der CPU befinden

3.7

Wozu dient der Kontextwechsel beim Umschalten zwischen Prozessen?

Fortsetzung eines Prozesses mit dem korrekten Programmzustand (Variablenbelegungen, etc.)

Fortsetzung eines Prozesses an der korrekten Stelle im Programmcode

3.8

Was sind Nachteile des kooperativen Umschaltens zwischen Prozessen?

Programmierer müssen selbstständig entscheiden an welchen Stellen im Prozess umgeschalten wird, was unter der Berücksichtigung sehr vieler Prozesse kompliziert ist.

"Gierige" Prozesse könnten den Prozessor für sehr lange Zeiträume belegen und die Ausführung anderer Prozesse somit verhindern.

3.9

Welche Aussage(n) ist/sind wahr für das präemptive Umschalten zwischen Prozessen?

Das Umschalten zwischen Prozessen wird von außen durch einen Wecker ausgelöst.

Prozesse haben keinen direkten Einfluss auf den Zeitpunkt des Umschaltens.

4.1 Kritischer Abschnitt

Welche Aussage(n) ist/sind wahr für kritische Abschnitte?

Kritische Abschnitte sind Codeabschnitte, die bei abwechselnder Ausführung zu Problemen führen können.

Kritische Abschnitte können nur auftreten, wenn verschiedene Programme die selben Datenstrukturen nutzen.

4.2

Was muss bei der Realisierung von gegenseitigem Ausschluss berücksichtigt werden?

Art der Umschaltung (kooperativ oder präemptiv)

Anzahl der Prozessoren

Vorhandensein von Interrupts

4.3

Warum ist es sinnvoll, Bedingungsvariablen für den gegenseitigen Ausschluss (Abschnittssperren) in einem atomaren Schritt zu überprüfen und zu setzen?

Das Überprüfen und Setzen ist selbst ein kritischer Abschnitt