

Fragen zur Wiederholung der Vorlesung IBN

Folgende Fragen sollen Ihnen beim Wiederholen des Stoffes zur Vorlesung IBN helfen.

Hinweise:

- Diese Fragen schließen andere Themen / Fragen in der Klausur nicht aus, aber wer diese Fragen sicher beantworten und die Antworten anwenden kann, wird die Klausur mit hoher Wahrscheinlichkeit mit mindestens „gut“ bestehen.
- In der Klausur wird nur zu einem Teil das Faktenwissen abgefragt, überwiegend muss man die sichere Anwendung und Verständnis dieses Wissens nachweisen. Deshalb ist es als Vorbereitung sehr hilfreich, die Konzepte an kleinen Beispielen selbst „nachzurechnen“ und zu überlegen, „was passiert, wenn“ (z.B. etwas schiefgeht, oder anders abläuft).

Teil A – Betriebssysteme

Grundlagen der Betriebssysteme

- Welche Aufgaben hat ein Betriebssystem?
- Was sind Caches und warum sind sie notwendig? Warum funktionieren sie überhaupt?
- Was passiert bei den Systemaufrufen (Trap, Parameterübergabe)?
- Was sind die Shells und was ist ihr Zweck im Betriebssystem?

Prozesse

- Was ist ein Prozess und wie unterscheidet sich dieser vom Programm?
- Was ist ein PCB und bei welchen Vorgängen wird er benötigt?
- Wie erzeugt und terminiert man Prozesse unter POSIX (fork, execve, waitpid, exit)?
- Welche Prozesszustände gibt es? Wie heißen diese und welche Bedeutung haben sie?
- Was triggert welche Übergänge dazwischen?

Interprozesskommunikation

- Welche Grundtypen der IPC gibt es und wie sind diese charakterisiert?
- Wie sieht konzeptionell eine Implementierung der IPC via shared memory aus (Systemaufrufe, was müssen sich die Prozesse noch mitteilen)?
- Was ist ein Pipe? Wie benutzt man (anonyme) Pipes in einer Shell?
- Wie benutzt man (anonyme) Pipes in POSIX / C?
- Was ist ein Dateideskriptor in POSIX?

Threads

- Was ist die Beziehung zwischen Threads und Prozessen? Welche Vorteile hat die Verwendung von Threads?
- Was sind die Unterschiede zwischen Kernel- und Userthreads?
- Was sind die Grundaufreufe zur Threadverwaltung in POSIX?
- Warum ist Threadprogrammierung schwierig? Was könnte z.B. bei dem Beispielpogramm der VL schiefgehen, wenn man die Parameter an die Threads anders übergibt? (Wie anders und warum ist das nicht gut)?

Race Conditions und Mutual Exclusion

- Was ist eine Race Condition (Definition)?
- Geben Sie ein Beispiel eines einfachen MT-Programms und einer Programmausführung an, bei der ein "lost update"-Problem auftritt.

- Wie kann man Race Conditions verhindern?
- Was ist der wechselseitige Ausschluss und was sind die Bedingungen dafür?
- Wie lautet die Lösung von Peterson für den wechselseitigen Ausschluss? Geben Sie ein Beispiel der Ausführung an.
- Welche Hardware-Instruktionen ermöglichen es, Locks zu implementieren, und wie funktioniert das genau (Befehle + Schleife)?
- Was sind Semaphore, insbesondere wie verhalten sich die beiden Operationen?
- Wie implementiert man wechselseitigen Ausschluss und Codeblock-Abhängigkeit mit Semaphoren?
- Wie kann man Semaphore ohne aktives Warten implementieren? Konzept und ggf. Pseudocode dazu.
- Was sind die condition variables, und wie verwendet man die POSIX-Funktion `*wait(condition, mut)`?

Grundlagen der Speicherverwaltung

- Wie funktioniert Swaping, und wozu ist es da?
- Was ist der Unterschied zwischen Swaping und Paging?
- Welche zwei Herausforderungen des Speichermanagements gibt es?
- Was ist die externe Fragmentierung, was ist die interne Fragmentierung?
- Was ist die MMU? Was ist/sind logischer Adressraum/logische Adressen versus physischer Adressraum/physische Adressen?
- Wie funktioniert eine einfache MMU mit Basis- und Limitregister? Welche der beiden Herausforderungen des Speichermanagements kann diese lösen?

Paging und Seitentabellen

- Geben Sie eine Definition des Pagings an. Welche Einschränkung hat dadurch die Übersetzung der Adressen, damit es umsetzbar ist?
- Wie wird eine logische Adresse aufgeteilt, und was ist die Seitennummer bzw. der Seitenindex, und der Seitenoffset?
- Was ist eine Seitenrahmennummer?
- Wie funktioniert die Adressenübersetzung auf der konzeptionellen Ebene, d.h. noch ohne eine Seitentabelle? Mit anderen Worten, welche Teile der logischen Adresse werden übersetzt und wodurch (hierbei soll man Begriffe wie Seitennummer, Seitenrahmennummer usw. verwenden)?
- Wie funktioniert das Prinzip der Adressenübersetzung durch eine (direkte) Seitentabelle (ein Diagramm als Antwort)? Was ist der Index, was ist der Inhalt eines Eintrags in der Seitentabelle?
- Wie hilft das Paging, die externe Fragmentierung zu beseitigen?
- Warum muss jeder Prozess eine eigene (direkte) Seitentabelle haben?
- Was ist ein TLB, und warum ist es sehr wichtig und sehr schnell?
- Wie berechnet man die effektive Zugriffszeit?
- Was ist die maximale Größe einer direkten Seitentabelle? Welche Daten muss man zur Berechnung haben?
- Was ist das Konzept einer invertierten Seitentabelle, und welche Daten enthält diese?
- Braucht man eine einzige invertierte Seitentabelle, oder eine pro Prozess?
- Wie viel Speicherplatz braucht eine invertierte Seitentabelle, und welche Daten benötigt man zur Berechnung? Diese Frage kann auch in der Variante gestellt werden: Welchen Anteil des RAM braucht eine invertierte Seitentabelle?
- Was ist das Konzept der mehrstufigen Seitentabellen? Insbesondere, welche Daten enthalten die einzelnen Stufen? (Hierbei soll man verstehen, dass nur die unterste Stufe einer direkten Seitentabelle entspricht, während alle höheren Stufen (page directory etc.) nur die Anfangsadressen der Stücke der tieferen Stufen enthalten).
- Hinweis zu der „Zusammenfassung: Paging und Seitentabellen“: Das sollte man sich sehr genau für die Klausur anschauen!

Virtuelle Speicherverwaltung und Seitenersetzung

- Was ist die virtuelle Speicherverwaltung, und was ist die Abgrenzung zum Paging?

- Was ist ein Seitenfehler, und wie merkt die Hardware bzw. das Betriebssystem, dass man einen erzeugen sollte?
- Wie berechnet man die effektive Zugriffszeit unter Berücksichtigung der Seitenfehler?
- Was ist das Demand Paging? In welchem Zusammenhang benutzt man Copy-On-Write?
- Was ist der prinzipielle Ablauf der Seitenersetzung?
- Was sind die M-Bit und R-Bit? Wann und wo werden sie gesetzt?
- Was ist die Bellady-Anomalie? (Eine Erklärung als Diagramm wird bevorzugt).
- Wie funktioniert der FIFO- Seitenersetzungsalgorithmus? Hier soll man imstande sein, ein Beispiel anzugeben.
- Dasselbe für den Second-Chance-Algorithmus.
- Wie funktioniert der erweiterte Clock-Algorithmus?
- Wie funktioniert der Least-Recently-Used (LRU)-Algorithmus? Wo liegt die Schwierigkeit der Implementierung?
- Wie kann man die Zeit des letzten Zugriffs mit den R-Bits approximieren?
- Was ist das Working set und wie definiert man diesen für die technische Nutzung?
- Wie kann man Working Set tatsächlich effizient annähern?

Segmentierung

- Was ist die Segmentierung und was muss der Programmierer dabei beachten?
- In welchen Aspekten und wie unterscheiden sich Paging und Segmentierung?

Dateien, Dateisysteme, Festplatten

- Welche Systemaufrufe / Bibliotheksfunktionen sind notwendig, um eine Datei (in C) zu kopieren?
- Wie unterscheiden sich harte und symbolische Links? Erläutern Sie das an einem Diagramm.
- Wie unterscheiden sich die CHS-Adressierung und die logische Adressierung der Blöcke einer Festplatte?
- Was sind die gravierenden Nachteile einer zusammenhängenden Belegung (bei der Speicherung von Dateien auf einer Festplatte)?
- Wie funktionieren bei der Speicherung von Dateien auf einer Festplatte a. Belegung durch verkettete Listen und b. Die Verbesserung durch FAT?
- Was ist das Konzept der I-Nodes? Welche Probleme des FAT- Ansatzes löst es?
- Wie speichert man mit den I-Nodes die Adressen der Blöcke bei sehr großen Dateien? Erläutern Sie das in Detail und mit einem Diagramm.
- In einem Dateisystem mit I-Nodes ist die maximale Länge einer Datei begrenzt. Wie können Sie die Obergrenze berechnen, und welche Angaben brauchen Sie dazu?
- Welche zwei Faktoren beeinflussen die Zeit bis zum Lesen des 1. Bytes bei einer rotierenden Festplatte?
- Wie funktionieren folgende Algorithmen zum Umräumen der Kopfbewegungen: SSTF, SCAN, C-SCAN?
- Wie funktioniert RAID 0 und RAID 1, und was ist der Zweck von jedem dieser beiden Systeme?
- Wie berechnet man die Ausfallwahrscheinlichkeit von RAID 0, und die von RAID 1, und welche Parameter sind für eine Berechnung notwendig?
- Wie funktioniert RAID 4 und welches Problem hat dieses System?
- Wie funktioniert RAID 5, und warum vermeidet es das Problem von RAID 4?

Deadlocks

- Was ist ein Deadlock? Hier ist die genaue Definition gefragt.
- Geben Sie ein möglichst einfaches Beispiel eines Deadlocks bei der falschen Verwendung von Semaphoren an.
- Erläutern Sie die vier Voraussetzungen für die Entstehung eines Ressourcen-Deadlocks.
- Wie können Sie die aktuelle Situation der Belegung der Ressourcen und Nachfragen nach ihnen grafisch darstellen? Wie heißt der zugehörige Graph?
- Falls es pro Ressourcentyp nur eine einzige Instanz einer Ressource gibt, wie äußert sich die Zyklische Wartebedingung in dem Graphen?

- Falls wir mehrere Instanzen pro Ressourcentyp haben, können wir anhand des oben genannten Graphen leicht erkennen, ob ein Deadlock vorliegt, oder nicht? Warum?
- Beschreiben Sie und geben Sie ein konsistentes Beispiel für die Datenstrukturen an: Aktuelle Belegungsmatrix, Anforderungsmatrix, Ressourcenvektor, Ressourcenrestvektor.
- Zu welchem Zweck dient der Bankieralgorithmus und wann sollen wir diesen einsetzen? Wann funktioniert etwas Einfacheres zum gleichen Zweck?
- Beschreiben Sie die Idee und geben Sie Pseudocode für den Bankieralgorithmus an.
- Was ist die zentrale Idee bei der Verhinderung von Deadlocks?
- Wie wird bei der Verhinderung von Deadlocks der Bankieralgorithmus verwendet?

Scheduling

- Welche zwei Typen von Prozessen in Hinblick auf die CPU-Verwendung unterscheidet man?
- Erläutern Sie die folgenden Scheduling- Strategien: FCFS, SJF, Shortest Remaining Time First.
- Für welche Arten von Systemen/Anwendungsszenarien werden diese drei Scheduling-Strategien verwendet?
- Welche drei Typen von Anwendungsszenarien unterscheidet man? Nennen Sie mind. je zwei Verfahren für die besprochenen zwei Typen (d.h. ohne „Echtzeit“)
- Wie funktioniert das Round-Robin-Scheduling, und welche Größe kann man dabei festlegen?
- Beschreiben Sie das Konzept des Prioritätsscheduling. Welche Ihnen bereits bekannte Technik ist ein Spezialfall davon?
- Erläutern Sie die Ideen des Lotterie-Schedulings.

Security

- Beschreiben Sie die Idee eines Puffer-Überlaufs-Angriffs. Erläutern Sie dabei, welche Daten auf dem Stack abgelegt werden.

Teil B – Netzwerke

Vorläufig, Inhalte werden noch an den tatsächlichen Vorlesungsstoff angepasst

N01

- Was bedeutet der Begriff Host, und wie wird das begründet?
- Was ist das Prinzip der Leitungsvermittlung, und das der Paketvermittlung? Was wird in den heutigen Computernetzwerken verwendet?
- Was sind die wesentlichen Vorteile, und die wesentlichen Nachteile der Paketvermittlung?
- Benennen Sie die möglichen Arten der Verzögerungen bei der Paketvermittlung.
- Welche Art der Verzögerung hängt mit der Bandbreite einer Leitung zusammen, welche mit der Entfernung zwischen Endpunkten der Übertragung?
- Welche Informationen ermittelt das Dienstprogramm traceroute?

N02

- Benennen Sie die fünf Schichten des Internet-Protokollstapels.
- Durch welche Endpunkte der Kommunikation wird jeder der Schichten identifiziert? Geben Sie weiterhin für jede Schicht mindestens ein Beispielprotokoll an.
- Welche dieser fünf Schichten implementieren jeweils: Hosts, Router, Switches?
- Wenn zwei Prozesse miteinander kommunizieren, die Komponenten einer Netz-Anwendung sind, wie erkennt man unter diesen den Client, und wie den Server (hier ist die Funktion gemeint, und nicht die Anwendungsarchitektur)?
- Was ist der „Unterschied“ zwischen WWW und Internet, sowie zwischen HTTP und HTML (Definition reicht hier, herausstellen der Unterschiede macht zum Teil wenig Sinn)?

- Welches Protokoll der Transportschicht benutzt HTTP? Was merkt sich der Server bzw. der Client zwischen den Anforderung/Antwort –Paaren?
- Geben Sie konzeptionell (grob) das Format einer HTTP-Anforderung und einer HTTP-Antwort an.
- Was sind Cookies und welchem Zweck dienen sie (Bezug zu einer früheren Frage?)? Erläutern Sie diese anhand eines Diagramms.

N03

- Warum gibt es überhaupt UDP, obwohl es keinen zuverlässigen Datentransfer garantiert?
- Was sind die vier Felder in dem Header eines UDP-Pakets?
- Was sind die Sockets und was sind die Ports?
- Welche zwei Typen von Sockets kann ein Anwendungsprogrammierer benutzen? Welche Typen von Protokollen der Transportschicht verwendet jeder dieser Typen?
- Beschreiben Sie konzeptionell (möglicherweise durch ein Diagramm), wie die Kommunikation zwischen Client und Server mit den UDP-Sockets funktioniert. Sie brauchen hier die Namen der API-Aufrufe nicht zu kennen.
- Was für eine Art Abstraktion liefert eine TCP-Verbindung? Welche Zusicherungen werden hier für den Anwendungsprogrammierer gemacht?
- Beschreiben Sie konzeptionell, was aus der Sicht des Anwendungsprogrammierers auf der Serverseite bei einer TCP-Verbindung passiert. Insbesondere: welche Art von Sockets werden verwendet, und was es ihr Funktion?
- Beschreiben Sie analog die Sicht des Anwenderprogrammierers auf der Clientseite (bei TCP). Wenn der Client mehrmals Daten an den Server sendet (über die gleiche Verbindung), wie macht der das, und welche Informationen muss er/muss er nicht jedes Mal mitgeben? Vergleichen Sie das auch mit UDP-Programmierung.
- Definieren Sie den Begriff Multiplexing und Demultiplexing.
- Wie funktioniert das Demultiplexing bei UDP? Insbesondere, welche Daten dienen als Kriterium für das Finden des richtigen Zielsockets?
- Wie funktioniert das Demultiplexing bei TCP? Auch hier, welche Daten dienen zum Auffinden des korrekten Zielsockets?

N04

- Benennen Sie drei von den sechs grundlegenden Eigenschaften des TCP-Protokolls.
- Neben der Portnummern der Quelle und des Ziels, enthält der Header eines TCP Segmentes die Sequenz-Nummer und die Acknowledgement-Nummer. Erläutern Sie an einem Diagramm die Funktion dieser beiden Nummern (z.B. telnet-Szenario).
- Beschreiben Sie kurz die drei Phasen des 3-Wege-Handshake. Welche Bits werden wann gesetzt, und welche Rollen spielen dabei die Sequenz-Nummern und die Acknowledgement-Nummern?
- Was ist die Flusssteuerung, und wie wird sie umgesetzt? Die Namen der Variablen sind nicht wichtig, nur das Konzept.
- Wie funktioniert das Stop-and-Wait-Protokoll für die verlässliche Nachrichtenzustellung?
- Was ist der gravierende Nachteil dieses Protokolls, und mit welcher kleinen Änderung kann man diesen Nachteil beseitigen?
- Was ist das wesentliche Verhalten des Selektive Repeat-Protokolls? Benennen Sie die wichtigste Regel für den Empfänger, und die wichtigste Regel für den Sender.
- Was ist das Sliding Window, und wann wird das linke Ende davon (mit kleinsten Sequenznummern) jeweils beim Empfänger und beim Sender verschoben?

N05

- Was ist das wesentliche Verhalten des Go-Back-N-Protokolls? Vergleichen Sie es kurz mit dem Selektive Repeat-Protokoll.
- Bei TCP-Protokoll: was ist die wichtigste Verhaltensregel des Empfängers?
- Illustrieren Sie an einem Beispiel, was der Empfänger macht, wenn eines von z.B. vier Paketen des Senders verloren geht?
- Wie kann der Sender ohne einen Timeout erkennen, dass ein Paket oder ein ACK verlorengegangen ist? Was ist das genaue Kriterium, und was macht der Sender, wenn dieses zutrifft?
- Welche Adressen enthält der Header eines IP-Datagramms, und wie groß sind diese?
- Erläutern Sie kurz, mit welchem Mechanismus ein IP-Datagramm in kleinere Stücke zerlegt werden kann, insbesondere welche Felder im Header spielen dabei eine Rolle?
- Was ist ein Interface? Wie viele IP-Adressen kann ein Interface haben (Minimum, Maximum)?
- Haben die IP-Adressen innerhalb eines Subnetzes irgendwelche Gemeinsamkeit? Falls ja, welche?
- In welche zwei Teile wird eine IP-Adresse unterteilt? Was ist der Zweck von jedem dieser Teile? Warum führt man überhaupt diese Unterteilung durch?
- Wie unterscheidet sich die klassenbasierte Adressierung von der klassenlosen Adressierung? Wie wird bei der letzten die Länge des Netzwerkteils angegeben?

N06

- Auf welche Art kann man innerhalb einer (großen) Institution eine Hierarchie von Subnetzen erzeugen? Welcher Teil der IP-Adresse spielt dabei eine Rolle, und wie wird dieser weiter unterteilt?
- Eine Institution hat ca 2^{16} (= 65536) IP-Adressen bekommen. Wenn man innerhalb dieser Institution 64 Subnetze erzeugen möchte, wie müsste man den Hostteil unterteilen, und wie viele IP-Adressen kann jedes Subnetz maximal haben?
- Was ist der Zweck einer Subnetzmaske? Geben Sie Beispiele für die beiden Schreibweisen zur Angabe dieser Maske an (d.h. Längenschreibweise und Maskenschreibweise).
- Wie kann man mithilfe einer Subnetzmaske bestimmen, ob der Empfänger eines Pakets im gleichen Subnetz wie der Sender liegt?
- Was ist die Hauptbeschäftigung eines Routers? Was macht so ein Gerät in seiner Freizeit?
- Die Einträge der Weiterleitungstabelle eines Routers codieren Bereiche von IP-Adressen. Wie sieht diese Kodierung in Detail aus? Verdeutlichen Sie sich an einem Beispiel, wie man mithilfe dieser Kodierung schnell feststellen kann, ob eine IP-Adresse in einem IP-Bereich (festgelegt durch einen Eintrag der Weiterleitungstabelle) liegt.
- Angenommen, ein Router leitet alle Pakete mit der Ziel IP-Adresse aus einem IP-Adressenbereich a.b.c.0 bis a.b.c.255 auf Ausgang 0. Können Sie erreichen, dass die Pakete mit den Ziel IP-Adressen aus dem Bereich a.b.c.0 bis a.b.c.7 auf Ausgang 1 (statt 0) ausgegeben werden? Falls ja, welche Regeln verwenden Sie hier? Verdeutlichen Sie sich das an einem konkreten Beispiel, insbesondere geben Sie die Einträge der Weiterleitungstabelle für diesen Router an.
- Für Fortgeschrittene: Können Sie (zusätzlich zu der obigen Konfiguration) erreichen, dass Pakete mit den Ziel IP-Adressen im Bereich a.b.c.8 bis a.b.c.15 auf Ausgang 2 weitergeleitet werden? Geben Sie dazu den konkreten Eintrag in der Weiterleitungstabelle an.
- Beschreiben Sie kurz das Hauptkonzept und die vier Phasen des DHCP-Protokolls.
- Welchen Zweck hat die Source-NAT? Beschreiben Sie, was passiert, wenn ein Paket des lokalen Netzwerks den NAT-Router verlässt, und was passiert, wenn eine Antwort auf dieses Paket zurückkommt.
- Wie wird das ICMP-Protokoll bei dem Tool traceroute verwendet?

N07

- Wie modelliert man ein Netzwerk aus der Sicht der Routingalgorithmen? Welche drei Bestandteile hat so ein Modell?
- Geben Sie den Pseudocode für den Dijkstra-Algorithmus an, und erläutern Sie die Bedeutung der einzelnen verwendeten Datenstrukturen.

- Wir haben bei unserer Variante des Algorithmus die Größe $p(v)$ eingeführt, die den letzten Knoten vor dem Ziel v auf einem momentan kostengünstigsten Pfad von der Quelle u zu v angibt. Wenn der Dijkstra-Algorithmus fertig ist (für die Quelle u), wie kann u seine eigene Weiterleitungstabelle aus $p(v)$ berechnen?
- Geben Sie die Bellmann-Ford-Gleichung zusammen mit der Beschreibung der relevanten Größen an.
- Angenommen, der Knoten u kennt für jeden seinen Nachbarn v einen kostengünstigsten Pfad (und seine Kosten) von v zu dem Zielknoten z . Wie könnte u die Bellmann-Ford-Gleichung verwenden, um die Kosten eines kostengünstigsten Pfades zum Zielknoten z zu berechnen, und welche Angaben muss u dazu noch kennen?
- Beschreiben Sie das Verhalten jedes Knoten, welcher an dem Distanzvektor-Routing-Algorithmus (DV-Algorithmus) teilnimmt (das eventuell als Pseudocode).
- Was sind die Endpunkte der Kommunikation in der Sicherungsschicht?

N08, N09 und ggf. weitere

- Was ist eine MAC-Adresse? Vergleichen Sie ihre Eigenschaften mit den Eigenschaften einer IP-Adresse.
- Wie bestimmt man die MAC-Adresse eines Hosts aus seiner IP-Adresse? Wie funktioniert das entsprechende Protokoll?
- Angenommen, zwei Subnetze S_1 , S_2 hängen an verschiedenen Ausgängen von demselben Router R , und Host A aus S_1 schickt einem Host B aus S_2 ein Paket. Beschreiben Sie an einem Beispiel die Rollen der IP-Adressen und der MAC-Adressen von A , B , und den Ports von R (d.h. was muss alles passieren, damit das Paket bei B ankommt?).
- Beschreiben Sie kurz das Konzept eines Verfahrens, das Ethernet benutzt, um Fehlertoleranz trotz möglicher Kollisionen auf einem gemeinsamen Medium zu erreichen.
- Welche Typen von Adressen benutzt ein Switch, um zu entscheiden, an welchen Ausgang ein Paket weitergeleitet werden soll?
- Woher erfährt der Switch die Zuordnung zwischen diesen Adressen und den richtigen Ports? Beschreiben Sie kurz das Verfahren des Lernens dieser Zuordnung.
- Angenommen, ein Host wird einem Subnetz hinzugefügt (via DHCP). Dieser Host möchte als erstes die IP-Adresse von <http://www.helene-fischer.de/> erfahren (Eine ganz tolle Seite übrigens! Kennen Sie auch die offizielle Mobile App? ☺). Welche Protokolle und Schritte laufen hier ab, und welche Typen von Adressen und welche Protokollschichten sind involviert (seit dem Anfang, das heißt wenn der Host noch gar nicht im Netzwerk ist)?