

1) Welche Zustände kann ein Prozess haben?

- 1.) Killed, Created, Ready to Run, Sleep
- 2.) Zombie, Created, Ready to Run , Sleep, Running**
- 3.) Forked , Zombie , Running, Sleep
- 4.) Getting created, Zombie,Ready to Run , Sleep, Running

2) Wie heißt eine Datei die nicht menschenlesbar ist und in Maschinensprache geschrieben ist?

- 1.) Binärdatei**
- 2.) Quellprogramm
- 3.) Objektprogramm**
- 4.) Textdatei aus Quellcode

3) Welche Möglichen Variablen Typen gibt es in C++ ?

- 1.) char**
- 2.) unsigned int**
- 3.) boolean
- 4.) unsigned float
- 5.) unsigned char**

4) FrameStack

- 1.) Der Frame Stack arbeitet nach dem FIFO Prinzip
- 2.) Bei Funktionsaufrufen wird der Frame auf den Frame Stack gelegt**
- 3.) Für jede Funktion wird ein Frame Stack erstellt**
- 4.) Der Framestack speichert Rücksprungadressen, Aufrufparameter und Lokale Variablen**

5) SystemCalls

- 1.) System Calls dienen zur Prozessverwaltung, Dateisystem/-verwaltung**
- 2.) durch den fork() Aufruf, wird ein anderer duplicierter Prozess erstellt, der gleichzeitig läuft und dieselbe PID hat wie der Vater Prozess
- 3.) der exec () Aufruf, überlädt den laufenden Prozess, inklusive allen threads, wobei die PID erhalten bleibt**
- 4.) pid_t wait (int *status), wartet bis ein Rückgabestatus vom Kind kommt und beendet dann mit dem Rückgabestatus das Programm**

6) Zeiger

- 1.) speichern Adressen eines bestimmten Speicherbereichs**
- 2.) Rekursive Datenstrukturen lassen sich nicht mit Zeigern erstellen
- 3.) haben keinen eigenständigen Typen, sondern sind vom jeweiligen Basistypen abgeleitete konstruierte Typen**

4.) Array- Variable ist Zeiger auf erstes Array-Element

7) Prozesse und Jobs

1.) eine Gruppe von Prozessen die von einem commando oder script ausgeführt wurden wird auch Job genannt.

2.) um einen Befehl oder Prozess sofort im Hintergrund laufen zu lassen , hängt man vorne ein & an den Befehl an. (& -mr [Befehl])

3.) Falls ein Job im Hintergrund läuft, ist er nicht an der Shell gebunden und hat somit kein Zugriff auf den Shell Input.

4.) Jeder Prozess hat eine eindeutige Prozessnummer und einen eigenen privaten Arbeitsbereich im Speicher

8) Was ist ein Interrupt Request?

1.) konkretes Signal einer Hardwarequelle als Aufforderung an CPU

2.) konkretes Signal einer Softwarequelle als Aufforderung an CPU

3.) konkretes Signal einer Hardwarequelle als Aufforderung an Prozess

9.) Beim Preemptive Multi-Tasking ...

1.) ...werden die Prozesse im Zeitschiebeverfahren ausgeführt

2.) ...erfolgt die Zuteilung der CPU an einzelne Funktionen innerhalb eines Prozesses

3.) ... werden Threads, also Objekte mit neuer laufender CPU-Versorgung erzeugt.

10.) Konstruktoren ...

1.) Können mit Parametern versehen werden

2.) Sind zuständig für das Erzeugen bzw. Auflösen von Objekten

3.) Werden über den Namen der Klasse deklariert und haben einen Rückgabewert.

11.) Eine Header Datei enthält...

1.) Variablendeklarationen

2.) Klassendeklarationen

3.) Funktionsdeklarationen

12.) Was ist der Unterschied zwischen einem Programm und einem Prozess

1.) Prozess administriert ein Programm

2.) Prozess, ist die Ausführung eines Programms

3.) Programm, Aufgabenbeschreibung, welche auf einem Computersystem ausgeführt werden kann

4.) Ein Programm ist ein Prozess

13.) Was ist ein Job in der Shell?

1.) An die Shell gebundene Hintergrundprozesse

- 2.) An die CPU gebundene Programme
- 3.) keins davon

14.) Was macht das Schichtenmodell und wie arbeitet es?

- 1.) Schichten vermitteln zwischen Software und Hardware**
- 2.) Jede Schicht kommuniziert mit der nächst höheren bzw. -tieferen Schicht**
- 3.) Nur so können Systemaufrufe im Virtuellen Dateisystem erfolgen

15.) Abort vs. Exit

- 1.) void abort() beendet ein Programm mit Rückgabewert
- 2.) void abort() ruft keinen Destruktor auf**
- 3.) exit ruft Destruktor für statische Objekte auf**

16.) C++ eignet sich besonders für ...

- 1.) ... die Echtzeitprogrammierung, da sie Hardwarenah ist.**
- 2.) ... alle Arten von Programmierung
- 3.) ... Anforderungen, wo die Effizienz keine große Rolle spielt.

17.) Beim Multi Threading ...

- 1.) ... wird eine effizientere Prozessauslastung erreicht.**
- 2.) ... wird eine höhere Systemzuverlässigkeit erreicht.**
- 3.) ... teilen sich Prozesse/Task CPU-Zeit, CPU springt zwischen den Prozessen.

18.) Wie viele Schnittstellen hat eine virtuelle Maschine ?

- 1.) Keine
- 2.) höchstens eine
- 3.) min. eine**

19.) Der Zugriff auf den Hauptspeicher eines Rechners ist etwa 5 Mal langsamer als der Zugriff auf den CPU Cache ?

- 1.) Ja
- 2.) Nein**

20.) int main(int arg, char* argv[]) Welche Information speichert argc?

- 1.) Anzahl an Methoden die in ihrem Programm vorhanden sind
- 2.) Anzahl an Rückgabewerten ihrer Main Methode
- 3.) Anzahl an Zeichenketten, die auf argv zeigt**