

**Aufgabe 1.1: Einfachauswahl-Fragen (18 Punkte)**

Bei den Multiple-Choice-Fragen in dieser Aufgabe ist jeweils nur **eine** richtige Antwort eindeutig anzukreuzen. Auf die richtige Antwort gibt es die angegebene Punktzahl.

Wollen Sie eine Multiple-Choice-Antwort korrigieren, streichen Sie bitte die falsche Antwort mit drei waagrechten Strichen durch (  ) und kreuzen die richtige an.

Lesen Sie die Frage genau, bevor Sie antworten!

- a) Der Speicher eines UNIX-Prozesses ist in Text-, Daten- und Stack-(Stapel-)Segment untergliedert. Welche Aussage zur Platzierung von Daten in diesen Segmenten ist richtig? 2 Punkte
- Globale Variablen liegen im Daten-Segment.
  - Lokale `static`-Variablen werden bei jedem Betreten der zugehörigen Funktion neu initialisiert.
  - Bei einem Aufruf von `malloc(3)` wird das Stack-Segment dynamisch erweitert.
  - Der Code von Funktionen wird zusammen mit den Variablen der Funktion im Stack-Segment abgelegt.
- b) Man unterscheidet Traps und Interrupts. Welche Aussage ist richtig? 2 Punkte
- Der Zugriff auf eine logische Adresse kann zu einem Trap führen.
  - Bei der mehrfachen Ausführung eines unveränderten Programms mit gleicher Eingabe treten Interrupts immer an den gleichen Stellen auf.
  - Der Zeitgeber (Systemuhr) unterbricht die Programmbearbeitung in regelmäßigen Abständen. Die genaue Stelle der Unterbrechungen ist damit vorhersagbar. Somit sind solche Unterbrechungen in die Kategorie *Trap* einzuordnen.
  - Ein Nebeneffekt eines Interrupt kann sein, dass das unterbrochene Programm abgebrochen wird.
- c) Ein *laufender* Prozess wird in den Zustand *blockiert* überführt. Welche Aussage passt zu diesem Vorgang? 2 Punkte
- Der Prozess terminiert.
  - Der Prozess wartet auf Daten von der Festplatte.
  - Es ist kein direkter Übergang von *laufend* nach *blockiert* möglich.
  - Der Scheduler bewirkt, dass der laufende Prozess durch einen anderen Prozess verdrängt wird.

- d) Was versteht man unter virtuellem Speicher? 2 Punkte
- Speicher, der nur im Betriebssystem sichtbar ist, jedoch nicht für einen Anwendungsprozess.
  - Speicher, der einem Prozess durch entsprechende Hardware (MMU) und durch Ein- und Auslagern von Speicherbereichen vorgespiegelt wird, aber möglicherweise größer als der verfügbare physikalische Hauptspeicher ist.
  - Unter einem virtuellen Speicher versteht man einen physikalischen Adressraum, dessen Adressen durch eine MMU vor dem Zugriff auf logische Adressen umgesetzt werden.
  - Virtueller Speicher ist eine Bezeichnung für die in der konkreten Hardwarekonfiguration nicht vorhandenen Speicherbereiche des physikalischen Adressraums.
- e) Welche der folgenden Aussagen zu statischem bzw. dynamischem Binden ist richtig? 2 Punkte
- Bei dynamischem Binden müssen zum Übersetzungszeitpunkt alle Adressbezüge vollständig aufgelöst werden
  - Beim statischen Binden werden alle Adressen zum Ladezeitpunkt aufgelöst
  - Bei dynamischem Binden können Fehlerkorrekturen in Bibliotheken leichter übernommen werden, da nur die Bibliothek selbst neu erzeugt werden muss. Programme, die die Bibliothek verwenden, müssen nicht neu kompiliert und gebunden werden.
  - Statisch gebundene Programmdateien sind kleiner als dynamisch gebundene, da mehrfach genutzte Funktionen in einer shared library abgelegt werden und nicht in die ausführbare Datei kopiert werden.
- f) Man unterscheidet die Begriffe *Programm* und *Prozess*. Welche der folgenden Aussagen zu diesem Themengebiet ist richtig? 2 Punkte
- Ein Programm kann immer nur von einem Prozess gleichzeitig ausgeführt werden.
  - Das Programm ist der statische Teil (Rechte, Speicher, etc.), der Prozess der aktive Teil (Programmmähler, Register, Stack).
  - Wenn ein Programm nur einen aktiven Ablauf enthält, nennt man diesen Prozess, enthält das Programm mehrere Abläufe, nennt man diese Threads.
  - Ein Prozess ist ein Programm in Ausführung - ein Prozess kann aber auch mehrere verschiedene Programme ausführen.

- g) Wie funktioniert Adressraumschutz durch Eingrenzung? 2 Punkte
- Der Lader positioniert Programme immer so im Arbeitsspeicher, dass unerlaubte Adressen mit nicht-existierenden physikalischen Speicherbereichen zusammenfallen.
  - Begrenzungsregister legen einen Adressbereich im logischen Adressraum fest, auf den alle Speicherzugriffe beschränkt werden.
  - Begrenzungsregister legen einen Adressbereich im physikalischen Adressraum fest, auf den alle Speicherzugriffe beschränkt werden.
  - Die MMU kennt die Länge der verschiedenen Segmente eines Programms und verhindert Speicherzugriffe über deren Grenzen hinaus.
- h) Welche Aussage über `fork()` ist richtig? 2 Punkte
- Das im aktuellen Prozess laufende Programm wird durch das angegebene Programm ersetzt.
  - Der an `fork()` übergebene Funktionszeiger wird durch einen neuen Thread im aktuellen Prozess ausgeführt.
  - `fork()` erzeugt einen neuen Kind-Prozess und startet darin das als Parameter angegebene Programm.
  - Dem Vater-Prozess wird die Prozess-ID des neu erzeugten Kind-Prozesses zurückgeliefert.
- i) Was passiert, wenn Sie versuchen, in einem C-Programm über einen ungültigen Zeiger auf Speicher zuzugreifen? 2 Punkte
- Der Compiler erkennt die problematische Code-Stelle und generiert Code, der zur Laufzeit bei dem Zugriff einen entsprechenden Fehler auslöst.
  - Beim Laden des Programms wird die ungültige Adresse erkannt und der Speicherzugriff durch einen Sprung auf eine Abbruchfunktion ersetzt. Diese Funktion beendet das Programm mit der Meldung "Segmentation fault".
  - Die MMU erkennt die ungültige Adresse bei der Adressumsetzung und löst einen Trap aus.
  - Das Betriebssystem erkennt die ungültige Adresse bei der Weitergabe des Befehls an die CPU (partielle Interpretation) und leitet eine Ausnahmebehandlung ein.

**Aufgabe 1.2: Mehrfachauswahl-Fragen (4 Punkte)**

Bei den Multiple-Choice-Fragen in dieser Aufgabe sind jeweils  $m$  Aussagen angegeben,  $n$  ( $0 \leq n \leq m$ ) Aussagen davon sind richtig. Kreuzen Sie **alle richtigen** Aussagen an. Jede korrekte Antwort in einer Teilaufgabe gibt einen halben Punkt, jede falsche Antwort einen halben Minuspunkt. Eine Teilaufgabe wird minimal mit 0 Punkten gewertet, d. h. falsche Antworten wirken sich nicht auf andere Teilaufgaben aus.

Wollen Sie eine falsch angekreuzte Antwort korrigieren, streichen Sie bitte das Kreuz mit drei waagrechten Strichen durch (~~☒~~).

Lesen Sie die Frage genau, bevor Sie antworten!

- a) Gegeben sei folgendes Programmfragment: 4 Punkte

```
static int a = 20160217;
int func(int x) {
    static int b = 1;
    int c;
    int *d = malloc(800);
    d[0] = 12;
    ++x;
    ++b;
    return 0;
}
```

Welche der folgenden Aussagen zu den Variablen im Programm sind richtig?

- `c` ist uninitialized und enthält einen zufälligen Wert.
- `func` kann aus anderen Modulen heraus aufgerufen werden.
- Auf den Wert von `a` kann von anderen Modulen aus zugegriffen werden.
- Die Anweisung `++x` ändert den Wert von `x` und beeinflusst somit den Aufrufer.
- `c` verliert beim Rücksprung aus `func` seine Gültigkeit.
- `b` hat nach der Ausführung von `func` immer den Wert 2.
- Die Anweisung `d[0] = 12;` kann einen *Segmentation fault* auslösen.
- `d` zeigt auf ein Array, in dem Platz für 800 Ganzzahlen vom Typ `int` ist



**// Funktion main()**

**// Befehlszeilenargumente prüfen und weitere allgemeine  
// Vorbereitungen**

**// Arbeiter-Threads starten**

**// Auf Beendigung aller Threads warten, Gesamtergebnis  
// ausgeben, aufräumen**

**// Ende Funktion main()**

A:





