

Technikerschule - Fachschule für
Maschinenbau-, Metallbau-, Informatik- und Elektrotechnik
der Landeshauptstadt München



Technikerprüfung 2016/17

Datenverarbeitungstechnik TE2b

Zeit : 150 Minuten

Klasse :

Name :

	Punkte:	Note :	Unterschriften:
Erstkorrektur			
Zweitkorrektur			

Teil 1 , ohne Unterlagen

Name :

1. Kreuzen Sie hier alle richtigen Aussagen an :

- Protected mode schützt Freigaben gegen unberechtigten Zugriff
- In Windows laufen Anwenderprogramme im User Mode
- Kooperatives Multitasking ist deterministisch
- In präemptiven Multitasking können keine Fehler wie deadlocks passieren
- Tasks gehen von blocked nach running wenn der Peripheriezugriff endet
- Wenn eine Task zu lange sleeping ist, geht sie nach blocked
- In einer race condition blockieren sich zwei tasks gegenseitig
- Ein deterministischer Scheduler macht Multitasking berechenbar
- Der Windows-Scheduler ist in Prioritätsstufe „Echtzeit“ deterministisch

- virtual memory benötigt zum Speicherzugriff einen Internetzugang
- Die MMU verwaltet das paging
- das Präsenzbit zeigt an, ob Seiten beschrieben sind
- Im working set befinden sich alle Software-tools zur Speicherverwaltung
- Ein paging error zeigt eine Fehlfunktion der Festplatte

- Domänencontroller betreiben die Active Directory - Datenbank
- Kerberos ist ein Teil des Freigabe-Rechtesystems
- Gruppenrichtlinien dienen zur Verwaltung von Active Directory
- Verzicht auf „CTRL / ALT / ENTF“ zum login ist eine Benutzerrichtlinie
- NTFS-Rechte sind bei non-dedicated Fileservern wirkungslos
- Windows Server 2008 ist ein dedicated server

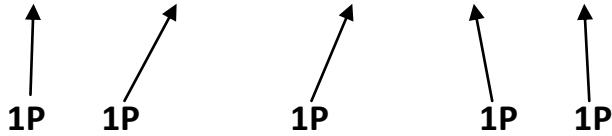
- Nach Korrektur eines fehlerhaften PHP-Skriptes benötigt Apache einen Restart
- HTML-Forms enthalten SQL-Befehle in der Form : mysql_query(-Befehl-)
- Bei redundanten Einträgen kann eine Datenbank abstürzen

23 Antworten : ¼ Punkt pro Antwort -> Rundung am Schluß (maximal 6 P.)

2. Geben Sie die korrekte URL zum Aufruf einer Webseite „hose.html“ an :

Server : hosenkauf
Domain : geheim.de
Port : 81
Seite : c:\webangebot\hosen\hose.html
Document-Root : c:\webangebot

http:// hosenkauf.geheim.de :81 / hosen/hose.html



(5 Punkte)

3. Die Seite „hose.php“ hat die beiden folgenden Tags am Anfang :

<html>
<body>

(Sie gleicht also eigentlich einer normalen HTML-Seite)

Was bewirkt aber die Fileextension **“.php“** dieser Seite ?

Apache ist so konfiguriert, daß er Seiten mit .php auf “<?php“ untersucht, und damit den Interpreter bedient. (Bei .html nicht !)

(3 Punkte)

4. Erklären Sie die Funktion des cgi-Mechanismus:

cgi lenkt Ausgaben von Skripten, die aus dem Web aufgerufen werden (meist php) zum Socket des Aufrufers um.

(3 Punkte)

5. Welcher Gedanke steckt hinter der working-set Strategie ?

Lokalitätsprinzip :

Adressbereich in Zukunft ist ähnlich dem Adressbereich aktuell

(3 Punkte)

Teil 2, mit Unterlagen

Erster Aufgabenteil : Windows-Skripting

Die Firma Pulverkönig führt Pulverbeschichtung an Produkten durch, die aus 3 Einzelteilen bestehen, welche in unterschiedlichen Farben beschichtet werden können. Wegen des Rüstaufwands werden nur Lose mit 50 Produkten (mit je 3 Einzelteilen) bearbeitet.

Die Aufträge werden in Form einer Liste losweise (immer 50) gespeichert. Jeder Kunde bestellt (vereinfacht) genau 1 Produkt.

(Der ganze Vorgang ist für die Prüfung stark vereinfacht)

Die Liste **Bestellliste.txt** sieht so aus :

```
bestell@tsm.de/Technikerschule/Rot/Rot/Blau  
Gerd.Pinsel@t-online.de/Pinsel/Rot/Blau/Blau  
Trump@donald.com/Trump/Blau/Blau/Blau
```

..und besteht (Losgröße) aus genau 50 Einträgen.

Zwei Farben sind möglich : Rot, Blau

Der Preis für das ganze Produkt besteht aus der Summe der Einzelteilpreise, welche sich je Los von 50 Stück so staffeln :

ab 1 Stück im Los: pro Stück 160 €
ab 10 Stück im Los: pro Stück 15 €
ab 50 Stück im Los: pro Stück 2 €

Aufgabe :

Erstellen Sie mit Hilfe von Windows Skripting aus der obigen Bestellliste eine neue Liste :

Kunde : bestell@tsm.de bezahlt : 32 Euro Kunde : Gerd.Pinsel@t-online.de bezahlt : 19 Euro Kunde : Trump@donald.com bezahlt : 6 Euro
--

(50 Zeilen..)

Gehen Sie bitte nach folgender Reihenfolge vor (einzelne Programmteile) :

- 1.1 Geben Sie den Programmkopf mit allen nötigen Einträgen an**
- 1.2 Durchsuchen Sie die gesamte Bestellliste für ein Los, und rechnen Sie aus, wie oft jede der Farben vorkommt.**
- 1.3 Ermitteln Sie den durch die Preisstaffel festgelegten Elementpreis für dieses Los.**
- 1.4 Schreiben Sie die Mailadresse und den Produktpreis in dem oben gezeigtem Format in die neue Liste mit dem Namen : Preisliste.txt**

1.1 Geben Sie den Programmkopf mit allen nötigen Einträgen an

```
Set Dateien = CreateObject("Scripting.FileSystemObject") (1 Punkt)
Set Eingabe = Dateien.OpenTextFile("C:\Users\user\Desktop\Bestelliste.txt", 1) ( 1 : 1 Punkt)
Set Ausgabe = Dateien.OpenTextFile("C:\Users\user\Desktop\Preisliste.txt", 8) ( 8 : 1 Punkt)
```

1.2 Durchsuchen Sie die gesamte Bestelliste für ein Los, und rechnen Sie aus, wie oft jede der Farben vorkommt.

```
Rotmenge = 0
Blaumenge = 0
while not Eingabe.AtEndOfStream (1 Punkt : Schleife ok)
    Zeile = Eingabe.Readline
    element = split(Zeile, "/") (1 Punkt Zeile lesen, untersuchen)
    if element(2) = "Rot" then (3 Punkte : Anzahl)
        Rotmenge = Rotmenge + 1
    else
        Blaumenge = Blaumenge + 1
    end if
    if element(3) = "Rot" then
        Rotmenge = Rotmenge + 1
    else
        Blaumenge = Blaumenge + 1
    end if
    if element(4) = "Rot" then
        Rotmenge = Rotmenge + 1
    else
        Blaumenge = Blaumenge + 1
    end if
wend
,-----
Eingabe.Close
Set Eingabe = Dateien.OpenTextFile("C:\Users\user\Desktop\Bestelliste.txt", 1) (1 Punkt)
'-----
```

1.3 Ermitteln Sie den durch die Preisstaffel festgelegten Elementpreis für dieses Los.

```
Rotpreis = 160
Blaupreis = 160
if Rotmenge > 10 then (3 Punkte : Staffel richtig)
    Rotpreis = 15
end if
if Rotmenge > 50 then
    Rorpreis = 2
end if
if Blaumenge > 10 then
    Blaupreis = 15
end if
if Blaumenge > 50 then
    Blaupreis = 2
end if
```

1.4 Schreiben Sie die Mailadresse und den Produktpreis in dem oben gezeigtem Format in die neue Liste mit dem Namen : Preisliste.txt

```
while not Eingabe.AtEndOfStream
  Zeile = Eingabe.Readline
  element = split(Zeile, "/")
  if element(2) = "Rot" then
    Preis = Preis + Rotpreis
  else
    Preis = Preis + Blaupreis
  end if
  if element(3) = "Rot" then
    Preis = Preis + Rotpreis
  else
    Preis = Preis + Blaupreis
  end if
  if element(4) = "Rot" then
    Preis = Preis + Rotpreis
  else
    Preis = Preis + Blaupreis
  end if
  Ausgabe.WriteLine("Kunde : "&element(0)&" bezahlt "&Preis&" Euro")
wend
```

(3 Punkte : jede Farbe mit Staffell)

(1 Punkt writeline)
(3 Punkte Format)

Zweiter Aufgabenteil: Webserver und Datenbanken

Um das Abschreiben unmöglich zu machen, sollen die Prüfungsaufgaben für jeden Schüler individuell aus einem großen Pool von Aufgaben zufällig ausgewählt werden.

Die dahinterstehende Datenbank soll zum Einen alle Prüfungsteilnehmer mit Namen, Vornamen und Email speichern, zum Anderen alle Aufgaben des Prüfungspools. Die Speicherung der Aufgaben geschieht vorab durch Speichern der Pfade zu den Aufgaben auf einem Fileserver (z.b. : `\\r2d2\filer\aufgabe35`). Diese Angabe wird einfach in einem Attribut als Text gespeichert.

2.1 :

a)

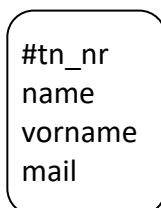
Entwerfen Sie eine Speicherstruktur (tables), in der unter Einhaltung aller nötigen Regeln für SQL-Datenbanken, die oben beschriebenen Inhalte gespeichert werden können. Stellen Sie hierzu die tables als ER-Modell dar. Es muß eine Beziehung zwischen Teilnehmern und Aufgaben herstellbar sein.

b)

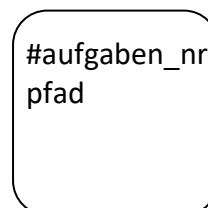
Geben Sie die table zur Speicherung der Aufgaben nun noch in Form der Tabelle an, mit zwei (frei erfundenen) Datensätzen als Beispiele.

a)

teilnehmer



aufgaben



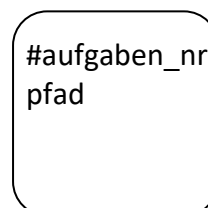
(je 2 Punkte)

ER-Prüfung : kommt 1 tn in aufgaben öfter vor ? -> n
kommt 1 aufgabe bein mehrern tn vor ? -> m

--> teilnehmer

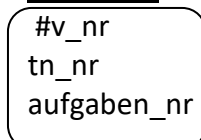


aufgaben



(2 Punkte)

verteilung



2.3

Mit den Werten aus der obigen Webseite wird am Server eine Aktion gestartet, die zunächst den Teilnehmer mit Name und Vorname und Mailadresse in die Teilnehmerliste einträgt.

Dann werden aus dem Aufgabenpool 10 Aufgaben zufällig ausgewählt.

Zur Erzeugung der zufälligen Auswahl dient ein php-Befehl, der Zufallszahlen erzeugt :

\$zufallszahl = **rand (1, 100)**; : dieser Befehl erzeugt z.B. eine Zufallszahl zwischen 1 und 100

Die ausgewählten Aufgaben werden dem Teilnehmer als Linkliste (im Browser) mitgeteilt, und es wird gespeichert, welche Aufgaben welcher Teilnehmer bekommen hat.

(Es müßte eigentlich noch sichergestellt werden, daß Aufgaben durch den rand-Mechanismus nicht doppelt gewählt werden, das lassen wir zur Vereinfachung aber auch weg)

```
<html>
<body>
Ihre Aufgabenliste :
<br><br>
<?php
    $t_name = $_POST["Famname"];          (1 Punkt : Variablen)
    $t_vor = $_POST["Vorname"];
    $t_mail = $_POST["Mail"];
    mysql_connect("localhost","root");    (1 Punkt : connect)
    mysql_select_db("tp2017");
    mysql_query("insert into teilnehmer ('name', 'vorname', 'Mail') values ('$t_name',
        '$t_vor', '$t_mail')");        (2 Punkte : insert)
    $result = mysql_query("select tn_nummer from teilnehmer where Mail = $t_mail ");
    while($row=mysql_fetch_array($result)) (2 Punkte : nummer fragen)
    {
        $tn = $row["tn_nummer"];
        for ($aufgabe=1; $aufgabe<11 ; $aufgabe++) (1 Punkt : Schleife)
        {
            $aufgabenummer = rand(1,100); (1 Punkt : rand)
            print "loesen Sie Aufgabe : ".$aufgabenummer; (1 Punkt : print)
            print "<br>";
            mysql_query("insert into liste ('teilnehmer', 'aufgabe') values ('$tn',
                '$aufgabenummer')"); (1 Punkt : Eintrag)
        }
    }
?>
</body>
</html>
```

(Summe : 10 Punkte)