

LFK Prozedurale Programmierung SS2012

Nur Screenshots der richtigen Antworten. Mit 60% bestanden.

Frage 1 von 40: Datenstrukturen - welche Aussagen treffen zu?

- Keine von den übrigen.
- Häufig verwendete Datenstrukturen sind Felder oder Arrays.
- Komplexere Datenstrukturen werden aus elementaren Typen aufgebaut.
- Datenstrukturen sind unübersichtlich und ihre Verwendung erschwert die Lesbarkeit von Programmen.
- Auf Elemente von Feldern kann über Indizes zugegriffen werden, deren Werte zur Laufzeit als Ausdrücke berechenbar sind.



Frage 4 von 40: In prozeduralen Programmiersprachen treten Operatoren und Funktionen auf. Durch welche Unterschiede sind sie gekennzeichnet?

- Funktionen und Operatoren weisen keine Unterschiede auf.
- In der Regel können zwar neue Funktionen, jedoch keine neuen Operatoren erstellt werden.
- Funktionen müssen in der Regel mit Argumenten von genau festgelegtem Typ aufgerufen werden; Operatoren dürfen mit Operanden verschiedener Typen verwendet werden.
- Funktionen besitzen im Unterschied zu Operatoren keine Präzedenz.
- Funktionen manipulieren ihre Argumente stets, Operatoren nie.



Frage 5 von 40: Operatoren (Präzedenz und Assoziativität) - welche Aussagen treffen zu?

- Präzedenzen und Assoziativitäten von Operatoren können bei verschiedenen Programmiersprachen unterschiedlich sein.
- Compiler, die das Prinzip Punkt- vor Strichrechnung beachten, arbeiten mit Operatoren unterschiedlicher Präzedenz.
- Keine von den übrigen.
- Ob ein Ausdruck mit gleichen Operatoren von links nach rechts oder umgekehrt evaluiert wird, hängt von der Assoziativität des Operators ab.
- Operatoren, die sowohl in binärer wie in unärer Form auftreten, können dabei unterschiedliche Präzedenzen aufweisen.



Frage 6 von 40: Prozedurale Programmiersprachen enthalten normalerweise Schlüsselwörter im definierten Sprachumfang. Außerdem bieten sie in der Regel die Möglichkeit, Bezeichner von Variablen und Unterprogrammen festzulegen. Welche Aussagen sind in diesem Zusammenhang richtig.

- Keine der übrigen Aussagen ist richtig.
- Im Sprachstandard der jeweilige prozeduralen Programmiersprache ist die Liste der reservierten Schlüsselwörter in der Regel festgelegt.
- Bei den meisten prozeduralen Programmiersprachen ist es ohne Einschränkung erlaubt, Schlüsselwörter als Bezeichner zu verwenden.
- Bei den meisten prozeduralen Programmiersprachen ist es möglich, Schlüsselwörter umzudefinieren und die damit freigewordenen Wörter als Bezeichner zu verwenden.
- Viele prozedurale Programmiersprachen verbieten die Verwendung von Schlüsselwörtern als Bezeichner.



Frage 7 von 40: Eine prozedurale Programmiersprache besitzt verschiedene Bestandteile. Welche gehören dazu?

- Speicherung von Daten in Variablen.
- Operatoren.
- Keine der übrigen.
- Garbage Collection.
- Ausführungssteuerung durch Fallunterscheidung und Schleife.



Frage 8 von 40: Bei der verteilten Entwicklung von prozeduralen Programmen ist die Versionsverwaltung besonders wichtig. Welches der folgenden Systeme ist zur verteilten Versionsverwaltung geeignet?

Keines von den übrigen.

Mercurial

Hydrargyrum

RCS

SCCS



Frage 9 von 40: Portabilität und Anbindung an Betriebssystem - welche Aussagen treffen zu?

Je umfangreicher in Programmen Betriebssystemfunktionen genutzt werden, desto geringer ist die Portabilität.

Das Design einer Programmiersprache kann mehr oder weniger umfangreiche Annahmen über Eigenschaften und Funktionen des Betriebssystems voraussetzen, unter dem die ausführbaren Programme schließlich laufen werden.

Portabilität ist bei der Erstellung von Programmen eine unwichtige Eigenschaft.

Betriebssystemfunktionen sollten über Standardschnittstellen aufgerufen werden, um eine möglichst hohe Portabilität zu erreichen.

Keine von den übrigen.



Frage 10 von 40: In vielen prozeduralen Programmiersprachen ist es möglich, Prototypen von Funktionen zu deklarieren. Welche Aussagen sind richtig?

Keine der übrigen Aussagen ist richtig.

Der Prototyp einer Funktion ist eine vorläufige Fassung des Quelltextes.

Prototypen deklarieren lediglich den Aufruf der Funktion.

Prototypen erlauben es dem Compiler, Anzahl und Typ der Argumente zu überprüfen.

Die Typprüfung der Argumente findet zur Laufzeit statt.



Frage 11 von 40: Was gilt bezüglich der Semantik einer prozeduralen Programmiersprache?

- Sie legt die Bedeutung der vom Programmierer verwendeten Bezeichner fest.
- Sie beschreibt, wovon die Wirkung syntaktischer Konstrukte abhängig ist.
- Keine der übrigen Antworten trifft zu.
- Sie legt die Bedeutung bzw. die Wirkung der syntaktischen Konstrukte fest.
- Sie listet die Teile der Sprache auf, die erst halb fertig entwickelt sind.



Frage 15 von 40: Die meisten prozeduralen Programmiersprachen bieten außer einfachen Datentypen auch aggregierte Datentypen an. Welche der folgenden Aussagen ist korrekt?

- Der Speicherbedarf eines aggregierten Datentyps ergibt sich aus der Anzahl der Komponenten und deren jeweiligem Typgröße sowie etwaigem zusätzlichem Alignment.
- Aggregierte Datentypen müssen immer aus Komponenten identischen Datentyps aufgebaut sein.
- Aggregierte Datentypen werden aus einfachen Datentypen aufgebaut.
- Keine der übrigen Aussagen ist korrekt.
- Komponenten aggregierter Datentypen können selbst aggregierte Datentypen sein.

Frage 17 von 40: Operatoren, die in prozeduralen Programmiersprachen verwendet werden, lassen sich nach unterschiedlichen Kriterien kategorisieren? Welche gehören dazu?

- Assoziativität.
- Existenz von Nebenwirkungen.
- Kardinalität.
- Präzedenz.
- Keines der übrigen.

Frage 18 von 40: In der Anwendungsentwicklung müssen prozedural erstellte Programme vielfach Betriebssystemfunktionen in Anspruch nehmen. Welche gehören dazu?

Funktionen der dynamischen Speicherverwaltung,

Funktionen zur Interaktion mit anderen Programmen,

keine der übrigen.

Funktionen der Ein- und Ausgabe,

Funktionen zur Bestimmung von Typgrößen,



Frage 22 von 40: Prozedurale Programmiersprachen können Sprachgruppen zugeordnet werden. Zu welchen der folgenden Kategorien zählen sie?

Objektorientierte Sprachen,

Imperative Sprachen,

Dynamische Sprachen,

Sie gehören zu keiner der übrigen.

Funktionale Sprachen,



Frage 23 von 40: Warum verwendet man in prozeduralen Programmen symbolische Konstante?

Literale können dadurch sprechende Namen erhalten, was die Verständlichkeit des Programms erhöht.

Die Compilation wird beschleunigt.

Aus keinem der übrigen Gründe.

Veränderung und Anpassung der Konstantenwerte werden erheblich erleichtert.

Das ausführbare Programm wird kleiner.

Frage 24 von 40: Der Begriff der "Deklaration" bezeichnet ein grundlegendes Konzept in der prozeduralen Programmierung. Welche Aussagen treffen zu?

- Eine Deklaration ist eine spezielle Willensbekundung, die der Programmierer in einem gesonderten Kommentar abgibt.
- Eine Deklaration entspricht einer Vereinbarung zur Festlegung von Namen und deren Typ.
- Mittels Deklaration wird dem Benutzer die Anwendung des Programms erläutert.
- Eine Deklaration legt den konkreten Wert des Bezeichneten fest.
- Keine der übrigen Aussagen ist korrekt.

Frage 25 von 40: Lokale Variable - welche Eigenschaften haben sie?

- Sie geben Auskunft über lokale Konventionen wie Sprache, Währung, Ortszeit und so weiter.
- Ihre Namen können in verschiedenen Funktionen oder Blöcken mehrfach genutzt werden.
- Keine von den übrigen.
- Sie enthalten die lokalen geographischen Koordinaten des Rechners.
- Sie sind nur in einem begrenzten Umfeld sichtbar.

Frage 26 von 40: Was gilt für Typen von Variablen und Literalen?

- Bei der Verarbeitung von Werten unterschiedlicher Typen muss gegebenenfalls eine implizite oder explizite Typwandlung erfolgen.
- Der exakte Speicherbedarf für einen Typ ist grundsätzlich durch den Standard der Programmiersprache festgelegt.
- Alle übrigen Aussagen sind falsch.
- In den meisten Programmiersprachen existieren elementare Typen wie Ganzzahlige und Fließpunkt.
- Typwandlung kann mit Informationsverlust behaftet sein.

Frage 27 von 40: Prozedurale Programmiersprachen setzen in der Regel gewisse Eigenschaften der Betriebssysteme voraus, auf denen die übersetzten Programme ausgeführt werden. Welche Aussagen treffen in diesem Zusammenhang zu?

- Die Kodierung von Zeichenkettenliteralen sollte derjenigen des Ausführungsbetriebssystems entsprechen.
- Rekursion und Laufzeitdimensionierung von Feldern setzt voraus, dass das Betriebssystem Methoden der dynamischen Speicherwaltung unterstützt.
- Dateizugriff setzt voraus, dass Dateien aus dem Programmkontext heraus identifizierbar sind.
- Bei der Programmierung kann angenommen werden, dass alle Betriebssysteme die gleiche interne Darstellung von Zeilenenden verwenden.
- Keine der übrigen Aussagen trifft zu.

Frage 28 von 40: In einigen prozeduralen Programmiersprachen ist eine Funktionsdeklaration möglich. Welche Aussagen sind in diesem Zusammenhang korrekt?

- Die Funktionsdeklaration beinhaltet eine implizite Funktionsdefinition.
- Keine der übrigen Aussagen ist korrekt.
- Die Funktionsdeklaration legt den Rückgabewert der Funktion fest.
- Die Funktionsdeklaration erlaubt dem Compiler eine formale Prüfung der korrekten Verwendung der Funktion.
- Die Funktionsdeklaration veranlasst den Compiler, Speicherplatz für die Funktion zu reservieren.



Frage 31 von 40: In vielen prozeduralen Programmiersprachen werden Operatoren unterschiedlicher Stelligkeit oder Kardinalität verwendet. Welche Aussagen sind zutreffend?

- Der Potenzoperator ist elitär.
- Der Additionsoperator ist in der Regel binär.
- Die Divisionsoperation ist singularär.
- Der Negationsoperator ist in der Regel unär.
- Keine der übrigen Aussagen trifft zu.



Frage 32 von 40: Welche Aussagen bezüglich der unbedingten Sprunganweisung ("Goto") treffen zu?

- Die Anweisung sollte nur für besondere Zwecke genutzt werden, zum Beispiel die Behandlung von Sondersituationen.
- Keine von den übrigen.
- Die häufige Verwendung dieser Anweisung kennzeichnet guten Programmierstil.
- Die Anweisung kann beliebig verwendet werden, weil sie auf Prozessorebene ebenfalls vorhanden ist.
- Die Anweisung sollte sparsam verwendet werden, weil durch sie der Programmfluss unübersichtlich werden kann.



Frage 33 von 40: Welche Bestandteile umfasst der Sprachumfang einer prozeduralen Programmiersprache im allgemeinen?

- Variable,
- Ein- und Ausgabe,
- Keine der übrigen.
- Operatoren,
- Kontrollstrukturen,



Frage 34 von 40: Dynamische Variable - welche Eigenschaften haben sie?

- Der von ihnen belegte Speicherplatz kann zwischen zwei Unterprogrammaufrufen anderweitig verwendet werden.
- Keine von den übrigen.
- Sie werden bei Unterprogrammaufrufen neu alloziert.
- Dynamische und statische Variable verhalten sich unterschiedlich.
- Ihre Inhalte können sich ohne Zutun des Prozessors verändern.



Frage 39 von 40: Welche der folgenden Systeme können zur Erstellung von Quelltexten prozeduraler Programme verwendet werden?

- Notepad.
- Paradox.
- Keines von den übrigen.
- Eclipse.
- Emacs.

