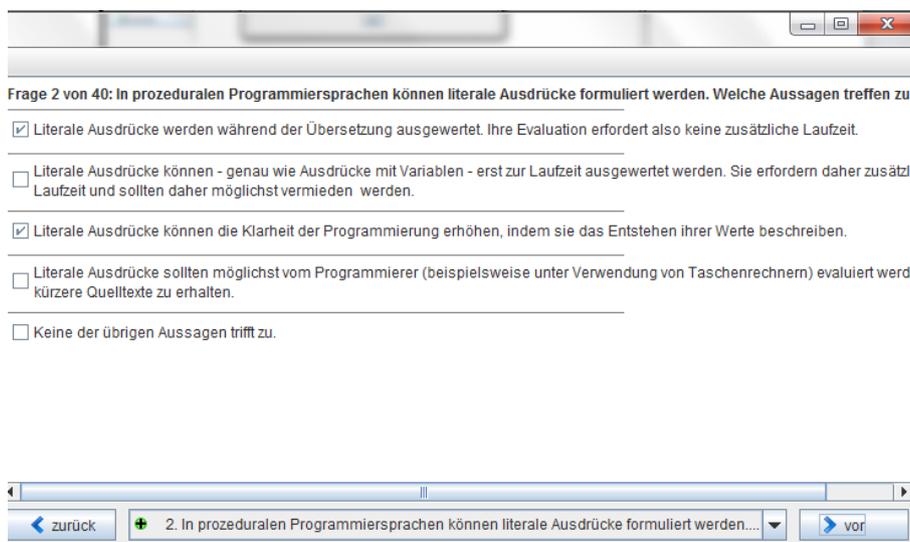
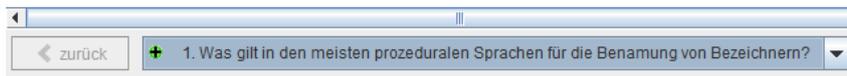


Frage 1 von 40: Was gilt in den meisten prozeduralen Sprachen für die Benennung von Bezeichnern?

- Keine der übrigen Aussagen trifft zu.
- Namen von Schlüsselwörtern dürfen als Bezeichner verwendet werden.
- Namen von Bezeichnern dürfen beliebig lang sein.
- Ein Name bildet jeweils ein Token.
- Das Anfangssymbol eines Bezeichners darf in der Regel keine Ziffer sein.



Frage 3 von 40: In prozeduralen Programmiersprachen werden Operatoren verwendet, die nicht nur einen Wert repräsentieren, sondern auch (Neben-)Wirkungen, sogenannte Seiteneffekte aufweisen. Welche der folgenden Operatoren gehören für die Programmiersprachen, in denen diese Operatoren unterstützt werden, dazu?

Subtraktionsoperator,

keiner der übrigen.

Dekrementoperator,

arithmetischer Vergleichsoperator,

Zuweisungsoperator,

zurück

3. In prozeduralen Programmiersprachen werden Operatoren verwendet, die nicht nur ...

vor

Frage 4 von 40: Beim Programmieren in prozeduralen Sprachen kann es geschehen, dass unerreichbarer Code erstellt wird. Welche Aussagen darüber sind richtig?

Viele Compiler erkennen unerreichbaren Code.

Unerreichbarer Code ist nachteilig hinsichtlich Codeumfang und -wartbarkeit.

Unerreichbarer Code gelangt niemals zur Ausführung, weil keine Kontrollstruktur den Programmablauf zu ihm führt.

Keine der übrigen Aussagen ist richtig.

Code wird sehr einfach durch unbedingte Sprungbefehle unerreichbar.

zurück

4. Beim Programmieren in prozeduralen Sprachen kann es geschehen, dass unerreich...

vor

Frage 6 von 40: Dynamische Variable - welche Eigenschaften haben sie?

Dynamische und statische Variable verhalten sich unterschiedlich.

Ihre Inhalte können sich ohne Zutun des Prozessors verändern.

Keine von den übrigen.

Der von ihnen belegte Speicherplatz kann zwischen zwei Unterprogrammaufrufen anderweitig verwendet werden.

Sie werden bei Unterprogrammaufrufen neu alloziert.

zurück

6. Dynamische Variable - welche Eigenschaften haben sie?

vor

**Frage 7 von 40: In der prozeduralen Programmierung findet ein Teil der dynamischen Speicherverwaltung über den Stack statt. Welche Aussagen sind korrekt?**

Keine der übrigen Aussagen ist korrekt.

Bei rekursiven Prozeduren erfolgt die Verwaltung von Zwischenwerten in der Regel auf dem Stack.

Globale Variable werden in der Regel im Stack gespeichert.

Beim Aufruf von Unterprogrammen werden die Rücksprungadressen in der Regel auf dem Stack abgelegt.

In der Regel werden Funktionsargumente auf dem Stack abgelegt.

zurück 7. In der prozeduralen Programmierung findet ein Teil der dynamischen Speicherverwa... vor

**Frage 8 von 40: Was gilt bezüglich der Phonetik einer prozeduralen Programmiersprache?**

Sie legt fest, bei welchem maximalen Hintergrundgeräusch prozedurale Programme der jeweiligen Sprache entwickelt werden können.

Keine der übrigen Antworten trifft zu.

Eine Phonetik ist für keine prozedurale Sprache festgelegt.

Sie legt die Aussprache der Wörter der Sprache fest.

Sie legt die Lautstärke des Piepers fest, mit dem auf Fehler hingewiesen wird.

zurück 8. Was gilt bezüglich der Phonetik einer prozeduralen Programmiersprache? vor

**Frage 9 von 40: Was gilt bezüglich der Pragmatik einer prozeduralen Programmiersprache?**

Sie erläutert, unter welchen Bedingungen die Sprache sinnvoll eingesetzt werden kann.

Sie beschreibt den bevorzugten Einsatzbereich der Sprache.

Sie grenzt die Einsatzbereiche ab, für die die jeweilige Sprache nicht optimal genutzt werden kann.

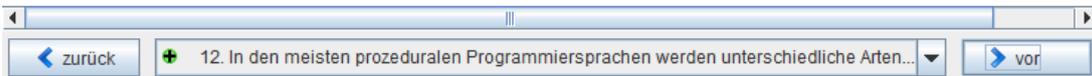
Keine der übrigen Antworten trifft zu.

Sie gibt dem Programmierer Hinweise, wie typische Probleme pragmatisch gelöst werden können.

zurück 9. Was gilt bezüglich der Pragmatik einer prozeduralen Programmiersprache? vor

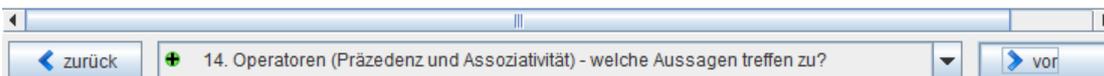
Frage 12 von 40: In den meisten prozeduralen Programmiersprachen werden unterschiedliche Arten von Schleifen unterstützt. Welche gehören dazu?

- abweisende Schleifen,
- zuweisende Schleifen,
- anweisende Schleifen,
- keine der übrigen.
- vorweisende Schleifen,



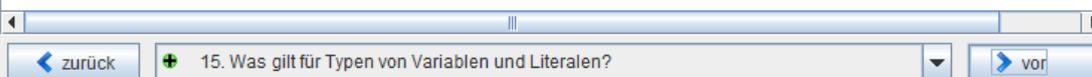
Frage 14 von 40: Operatoren (Präzedenz und Assoziativität) - welche Aussagen treffen zu?

- 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.
- Compiler, die das Prinzip Punkt-vor Strichrechnung beachten, arbeiten mit Operatoren unterschiedlicher Präzedenz.
- Präzedenzen und Assoziativitäten von Operatoren können bei verschiedenen Programmiersprachen unterschiedlich sein.
- Operatoren, die sowohl in binärer wie in unärer Form auftreten, können dabei unterschiedliche Präzedenzen aufweisen.



Frage 15 von 40: Was gilt für Typen von Variablen und Literalen?

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



**Frage 17 von 40: Schleifen bieten in prozeduralen Programmen oft Optimierungspotentiale. Welche Aussagen treffen in diesem Zusammenhang zu?**

- Schleifen können zur Steigerung der Verarbeitungsgeschwindigkeit ganz oder teilweise aufgelöst werden (Loop Unwinding). Ein typisches Beispiel in C ist "Duff's Device".
- Anweisungen, die von der Schleifensteuerung unabhängig sind, sollten aus der Schleife herausgezogen werden.
- Schleifen mit identischer Steuerung können und sollten zusammengefasst werden.
- Keine der übrigen Aussagen trifft zu.
- Ersetzen der Schleifensteuerung durch Rekursion steigert in der Regel die Ausführungsgeschwindigkeit.

zurück + 17. Schleifen bieten in prozeduralen Programmen oft Optimierungspotentiale. Welche ... vor

**Frage 18 von 40: Das Übersetzen eines Quelltextes ist die Aufgabe eines Compilers. Welche Aussagen über seine Funktionen sind korrekt?**

- Der Compiler erzeugt maschinenlesbaren Objektcode.
- Der Compiler übersetzt Kommentare ins Englische.
- Der Compiler überprüft den Aufbau des Quelltextes auf korrekte Syntax.
- Keine der übrigen Aussagen ist korrekt.
- Der Compiler zerlegt den Quelltext in Token und identifiziert diese.

zurück + 18. Das Übersetzen eines Quelltextes ist die Aufgabe eines Compilers. Welche Aussa... vor

**Frage 19 von 40: Welche Aussagen bezüglich Rekursion sind zutreffend?**

- Rekursiv programmierte Prozeduren weisen oft relativ lange Laufzeiten auf.
- Rekursion lässt sich immer auf Iteration zurückführen.
- Die übrigen Aussagen sind sämtlich falsch.
- Rekursion bewirkt häufig relativ großen Speicherbedarf.
- Rekursive Prozeduren lassen sich oft elegant und einfach programmieren.

zurück + 19. Welche Aussagen bezüglich Rekursion sind zutreffend? vor

Frage 20 von 40: Die Eigenschaften und die Verarbeitungsfähigkeit von Daten werden durch ihren Datentyp festgelegt. Welche Aussagen sind in Bezug auf prozedurale Programmiersprachen richtig?

- Der Datentyp eines Wertes wird stets erst zur Laufzeit festgelegt.
- Der Datentyp eines Wertes kann implizit oder explizit festgelegt werden.
- Keine der übrigen Aussagen ist richtig.
- Durch die Deklaration eines Wertes als bestimmter Datentyp wird der Umfang des dadurch belegten Speichers festgelegt.
- Die Menge der Werte eines Datentyps ist praktisch unlimitiert.

zurück + 20. Die Eigenschaften und die Verarbeitungsfähigkeit von Daten werden durch ihren D... vor

Frage 23 von 40: In prozeduralen Programmen sind häufig Typwandlungen (type casts) erforderlich. Welche Aussagen treffen zu?

- Datenwerte können bei Typwandlungen unter Umständen eine erhebliche Wertveränderung erfahren.
- Typwandlungen verursachen Laufzeitfehler.
- Man unterscheidet explizite und implizite Typwandlungen.
- Bei Typwandlungen kann es zu Genauigkeitsverlusten kommen.
- Keine der übrigen Aussagen trifft zu.

zurück + 23. In prozeduralen Programmen sind häufig Typwandlungen (type casts) erforderlich. ... vor

Frage 24 von 40: Welche Kontrollstrukturen werden in der prozeduralen Programmierung genutzt?

- Permutation.
- Keine von den übrigen.
- Alternation.
- Iteration.
- Sequenz.

zurück + 24. Welche Kontrollstrukturen werden in der prozeduralen Programmierung genutzt? vor

Frage 26 von 40: Was gilt bei Funktionsaufrufen in prozeduralen Programmen für Variablenübergabe per Call-by-Reference?

- Die übrigen Antworten sind sämtlich falsch.
- Die Variableninhalte werden per Zeiger übergeben.
- Die gerufene Funktion kann die Variableninhalte der rufenden Funktion verändern.
- Die Variablenübergabe per Call-by-Reference birgt das Risiko der Null-Pointer-Verfolgung.
- Der Zeit- und Speicheraufwand der Variablenübergabe ist von der Größe der übergebenen Datenstrukturen abhängig.

zurück 26. Was gilt bei Funktionsaufrufen in prozeduralen Programmen für Variablenübergab... vor

Frage 27 von 40: Zur Steuerung des Programmablaufs in Schleifen stellen viele prozedurale Programmiersprachen besondere Konstruktionen zur Verfügung. Welche Aussagen darüber sind richtig?

- Viele Sprachen unterstützen Anweisungen zum Verlassen der Schleife. Die Ausführung wird hinter der Schleife fortgesetzt.
- Keine der übrigen Aussagen ist richtig.
- Bei geschachtelten Schleifen bedingen Anweisungen zum Verlassen der innersten Schleife stets das Verlassen auch der übergeordneten Schleifen.
- Viele Sprachen unterstützen Anweisungen Überspringen des Schleifenrestes. Die Verarbeitung wird mit der nächsten Schleife fortgesetzt.
- Viele Sprachen unterstützen Sprungbefehle in eine Schleife, um beim ersten Schleifendurchlauf den Beginn der Schleife zu überspringen.

zurück 27. Zur Steuerung des Programmablaufs in Schleifen stellen viele prozedurale Progra... vor

Frage 29 von 40: In vielen prozeduralen Programmiersprachen werden Operatoren unterschiedlicher Stelligkeit oder Kardinalität verwendet. Welche Aussagen sind zutreffend?

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

zurück 29. In vielen prozeduralen Programmiersprachen werden Operatoren unterschiedliche... vor

Frage 30 von 40: In vielen prozeduralen Programmiersprachen können Ausdrücke unter Verwendung binärer Operatoren in der nachfolgend dargestellten Weise aufgebaut werden. In welchen dieser Ausdrücke beeinflusst die Präzedenz der Operatoren den Wert des Ausdrucks?

bei keinem der übrigen Ausdrücke.

$x = y == z$

$(2 + p) * q$

$a + 4 * b$

$24 / 3 / 2$

zurück 30. In vielen prozeduralen Programmiersprachen können Ausdrücke unter Verwendung... vor

Frage 31 von 40: Abhängig von der Anzahl der Argumente, auf die ein Operator wirkt, unterscheidet man unäre, binäre und ternäre Operatoren. Welche Aussagen treffen zu?

Unäre Operatoren sind in der Regel kommutativ.

Ternäre Operatoren liefern abhängig von einer Fallunterscheidung unterschiedliche Werte.

In vielen prozeduralen Programmiersprachen werden für unäre und binäre Operatoren dieselben Symbole verwendet.

Keine der übrigen Aussagen ist richtig.

Binäre Operatoren können nur bitweise angewendet werden.

zurück 31. Abhängig von der Anzahl der Argumente, auf die ein Operator wirkt, unterscheidet ... vor

Frage 32 von 40: Welche Arten von Operatoren treten in prozeduralen Programmiersprachen auf?

logische,

keine der übrigen.

arithmetische,

relationale,

administrative,

zurück 32. Welche Arten von Operatoren treten in prozeduralen Programmiersprachen auf? vor

Frage 34 von 40: In den meisten prozeduralen Programmiersprachen sind lokale Variable vorgesehen. Welche Aussagen sind in diesem Zusammenhang korrekt?

- Eine lokale Variable darf den gleichen Namen haben wie eine globale Variable. Diese ist dann im Kontext der lokalen Variablen nicht sichtbar.
- Lokale Variable sind nur im umgebenden Kontext ihres Prozedurblocks sichtbar.
- Wenn eine lokale Variable den gleichen Namen wie eine globale Variable besitzt, muss sie auch vom gleichen Typ sein.
- Keine der übrigen Aussagen ist korrekt.
- Alle lokalen Variablen eines Programms, die den gleichen Namen haben, belegen denselben Speicherplatz.

zurück

34. In den meisten prozeduralen Programmiersprachen sind lokale Variable vorgesehen...

vor

Frage 36 von 40: Eine prozedurale Programmiersprache besitzt verschiedene Bestandteile. Welche gehören dazu?

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

zurück

36. Eine prozedurale Programmiersprache besitzt verschiedene Bestandteile. Welche ...

vor

Frage 38 von 40: Welche Aussagen über die Auswertungsreihenfolge von Ausdrücken in prozeduralen Programmiersprachen treffen zu?

- Wenn in einer Programmiersprache die Punkt-vor-Strich-Regel gilt, dann ist für eine Summe von zwei Produkten die zeitliche Abfolge der Auswertung der Produkte in der Regel nicht vorhersagbar.
- Keine der übrigen Aussagen trifft zu.
- Wenn in einer Programmiersprache die Punkt-vor-Strich-Regel gilt, ergibt sich dies aus der Präzedenz der arithmetischen Operatoren.
- Die Auswertungsreihenfolge hängt von Präzedenz und Assoziativität ab.
- Ausdrücke werden grundsätzlich von links nach rechts ausgewertet.

zurück

38. Welche Aussagen über die Auswertungsreihenfolge von Ausdrücken in prozedural...

vor

Frage 40 von 40: Die meisten prozeduralen Sprachen kennen Schlüsselwörter. Was gilt in den meisten Sprachen bezüglich dieser Schlüsselwörter?

- Der Umfang der Schlüsselwörter ist definiert.
- Schlüsselwörter dienen zum Verschlüsseln vertraulicher Informationen.
- Der Programmierer kann eigene Schlüsselwörter ergänzen.
- Keine der übrigen Antworten trifft zu.
- Die Verwendung von Schlüsselwörtern als Bezeichnernamen ist erlaubt.



← zurück

+ 40. Die meisten prozeduralen Sprachen kennen Schlüsselwörter. Was gilt in den meis... ▼

→ vor