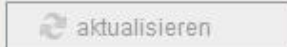
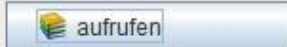


Menu

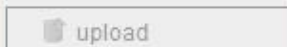
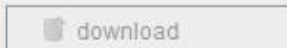


Aktionen:

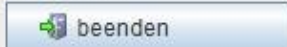
Projektübersicht:



Projekt:



Programm:



Frage 1 von 40: Welche Kontrollstrukturen werden in der prozeduralen Programmierung genutzt?

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

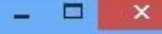


+ 1. Welche Kontrollstrukturen werden in der prozeduralen Programmierung genutzt?



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext

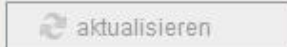
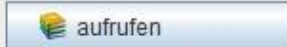


Menu

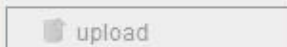


Aktionen:

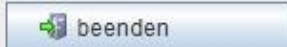
Projektübersicht:



Projekt:

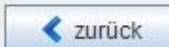


Programm:



Frage 2 von 40: Welche Arten von Operatoren treten in prozeduralen Programmiersprachen auf?

- Logische.
- Relationale.
- Keine der übrigen.
- Administrative.
- Arithmetische.

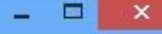


+ 2. Welche Arten von Operatoren treten in prozeduralen Programmiersprachen auf?



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext

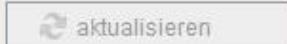
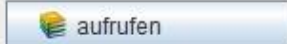


Menu



Aktionen:

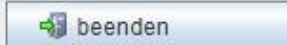
Projektübersicht:



Projekt:

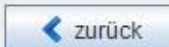


Programm:



Frage 3 von 40: Zu den Kontrollstrukturen der prozeduralen Programmierung gehören auch die Sprunganweisungen. Welche Aussagen sind in diesem Zusammenhang richtig?

- Man unterscheidet in der prozeduralen Programmierung bedingte und unbedingte Sprunganweisungen.
- Keine der übrigen Aussagen ist richtig.
- Bei bedingten Sprunganweisungen muss vor Ausführung des Sprungs die Rücksprungadresse gespeichert werden.
- Ein Sprung erfolgt jeweils zu einer entsprechend definierten Sprungmarke (Label).
- Sprunganweisungen sind in der prozeduralen Programmierung zwar prinzipiell vermeidbar, stellen jedoch in einigen Situationen eine effiziente Lösung dar, zum Beispiel beim Verlassen tief geschachtelter Schleifen.

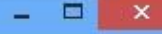


+ 3. Zu den Kontrollstrukturen der prozeduralen Programmierung gehören auch die Sprunganweisung...



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext

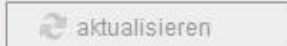
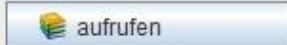


Menu

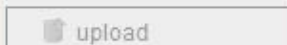


Aktionen:

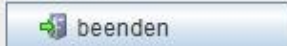
Projektübersicht:



Projekt:

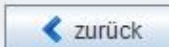


Programm:



Frage 4 von 40: Die meisten prozeduralen Programmiersprachen bieten Felder (Arrays) an. Welche Aussagen sind in diesem Zusammenhang korrekt?

- Keine der übrigen Aussagen ist korrekt.
- Die Veränderung einer Feldgröße zur Laufzeit ist grundsätzlich ausgeschlossen.
- Die Größe des Feldes ist grundsätzlich im Feld selbst hinterlegt.
- Die Elemente eines Feldes liegen im Speicher angeordnet vor, so dass sie per Index der Reihenfolge nach verarbeitet werden können.
- Die Elemente eines Feldes sind vom gleichen Datentyp.

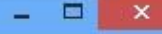


+ 4. Die meisten prozeduralen Programmiersprachen bieten Felder (Arrays) an. Welche Aussagen sin...



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext

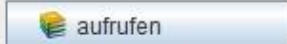


Menu

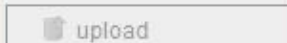
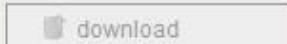


Aktionen:

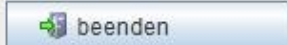
Projektübersicht:



Projekt:

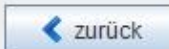


Programm:



Frage 5 von 40: Das Übersetzen eines Quelltextes ist die Aufgabe eines Compilers. Welche Aussagen über seine Funktionen sind korrekt?

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



+ 5. Das Übersetzen eines Quelltextes ist die Aufgabe eines Compilers. Welche Aussagen über seine ...



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext

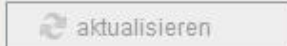
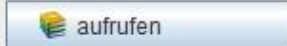


Menu



Aktionen:

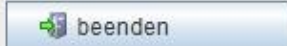
Projektübersicht:



Projekt:

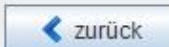


Programm:



Frage 6 von 40: Portabilität und Anbindung an Betriebssystem - welche Aussagen treffen zu?

- Portabilität ist bei der Erstellung von Programmen eine unwichtige Eigenschaft.
- Je umfangreicher in Programmen Betriebssystemfunktionen genutzt werden, desto geringer ist die Portabilität.
- Keine von den übrigen.
- Betriebssystemfunktionen sollten über Standardschnittstellen aufgerufen werden, um eine möglichst hohe Portabilität zu erreichen.
- 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.

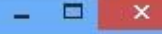


+ 6. Portabilität und Anbindung an Betriebssystem - welche Aussagen treffen zu?



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext

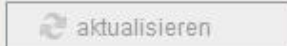
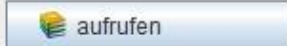


Menu



Aktionen:

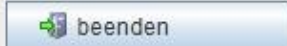
Projektübersicht:



Projekt:

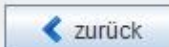


Programm:



Frage 7 von 40: Warum verwendet man in prozeduralen Programmen symbolische Konstante?

- Literale können dadurch sprechende Namen erhalten, was die Verständlichkeit des Programms erhöht.
- Das ausführbare Programm wird kleiner.
- Aus keinem der übrigen Gründe.
- Die Compilation wird beschleunigt.
- Veränderung und Anpassung der Konstantenwerte werden erheblich erleichtert.



+ 7. Warum verwendet man in prozeduralen Programmen symbolische Konstante?



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext



Menu



Aktionen:

Projektübersicht:

aufrufen

aktualisieren

Projekt:

bearbeiten

speichern

download

upload

Programm:

beenden

Frage 8 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?

 $24 / 3 / 2$ Bei keinem der übrigen Ausdrücke. $x = y == z$ $a + 4 * b$ $(2 + p) * q$ 

zurück



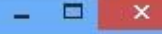
8. In vielen prozeduralen Programmiersprachen können Ausdrücke unter Verwendung binärer Operat...



vor

Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext

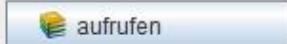


Menu

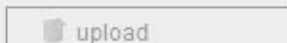


Aktionen:

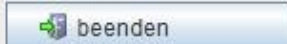
Projektübersicht:



Projekt:

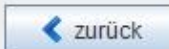


Programm:



Frage 9 von 40: Die Assoziativität von Operatoren ist in den prozeduralen Programmiersprachen ein wichtiges Konzept. Welche Aussagen sind in diesem Zusammenhang korrekt?

- Keine der übrigen Aussagen ist korrekt.
- Die Assoziativität der Operatoren kann sich erst auswirken, wenn mindestens zwei Operatoren in einem Ausdruck auftreten.
- Man unterscheidet links- und rechtsassoziative Operatoren.
- Assoziative Operatoren sind kommutativ.
- Die Assoziativität kann sich nur bei Operatoren der gleichen Präzedenz auswirken.

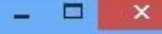


+ 9. Die Assoziativität von Operatoren ist in den prozeduralen Programmiersprachen ein wichtiges Kon...



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext



Menu



Aktionen:

Projektübersicht:

aufrufen

aktualisieren

Projekt:

bearbeiten

speichern

download

upload

Programm:

beenden

Frage 10 von 40: Manche prozeduralen Programmiersprachen wie Modula-2 unterstützen das Konzept der Coroutinen. Welche Aussagen treffen zu?

- Coroutinen können mehrere verschiedene Ein- und Ausstiegspunkte besitzen.
- Die Coroutine stellt eine Verallgemeinerung der klassischen Subroutine dar.
- Coroutinen sind geeignet, um kooperatives Multitasking zu implementieren.
- Coroutinen benötigen jeweils eine eigene Zustandsverwaltung. Deshalb sind sie in Sprachen, die nur einen zentralen Stack unterstützen, schwierig zu implementieren.
- Keine der übrigen Aussagen trifft zu.

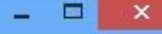
zurück

10. Manche prozeduralen Programmiersprachen wie Modula-2 unterstützen das Konzept der Corouti...

vor

Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext

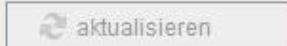
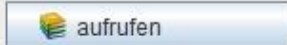


Menu



Aktionen:

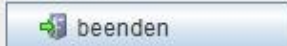
Projektübersicht:



Projekt:

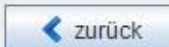


Programm:



Frage 11 von 40: In der prozeduralen Programmierung findet ein Teil der dynamischen Speicherverwaltung über den Stack statt. Welche Aussagen sind korrekt?

- In der Regel werden Funktionsargumente auf dem Stack abgelegt.
- Beim Aufruf von Unterprogrammen werden die Rücksprungadressen in der Regel auf dem Stack abgelegt.
- Bei rekursiven Prozeduren erfolgt die Verwaltung von Zwischenwerten in der Regel auf dem Stack.
- Keine der übrigen Aussagen ist korrekt.
- Globale Variable werden in der Regel im Stack gespeichert.

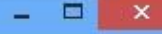


+ 11. In der prozeduralen Programmierung findet ein Teil der dynamischen Speicherverwaltung über d...



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext

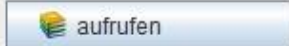


Menu

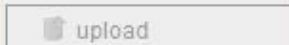
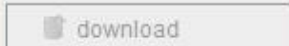


Aktionen:

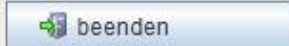
Projektübersicht:



Projekt:

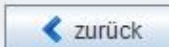


Programm:



Frage 12 von 40: Zur Steuerung des Programmablaufs in Schleifen stellen viele prozedurale Programmiersprachen besondere Konstruktionen zur Verfügung. Welche Aussagen darüber sind richtig?

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

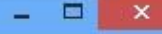


+ 12. Zur Steuerung des Programmablaufs in Schleifen stellen viele prozedurale Programmiersprache...



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext



Menu

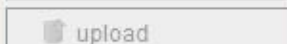


Aktionen:

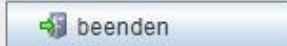
Projektübersicht:



Projekt:

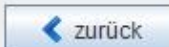


Programm:



Frage 13 von 40: In der prozeduralen Programmierung können Sprunganweisungen verwendet werden. Welche Aussagen sind in diesem Zusammenhang zutreffend?

- Ein Sprung, der zu einer Sprungmarke in eine Kontrollstruktur hinein führt, ist in der Regel erlaubt.
- Ein Sprung aus einer Kontrollstruktur heraus ist in der Regel erlaubt.
- Sprünge innerhalb einer Prozedur sind in der Regel erlaubt.
- Ein Sprung von einer Prozedur zu einer Sprungmarke in einer anderen Prozedur ist in der Regel erlaubt.
- Keine der übrigen Aussagen ist zutreffend.

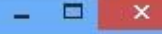


13. In der prozeduralen Programmierung können Sprunganweisungen verwendet werden. Welche ...



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext



Menu



Aktionen:

Projektübersicht:

aufrufen

aktualisieren

Projekt:

bearbeiten

speichern

download

upload

Programm:

beenden

Frage 14 von 40: In prozeduralen Programmen sind in der Regel nicht alle Variable in allen Programmteilen sichtbar. Welche der folgenden Aussagen sind in diesem Zusammenhang zutreffend?

Globale Variable bleiben auch in solchen Prozeduren sichtbar, in denen namensgleiche lokale Variable verwendet werden.

Keine der übrigen Aussagen ist zutreffend.

Wird ein Variablenname in mehr als einer Prozedur verwendet, dann wird die damit bezeichnete Variable global und damit im gesamten Programm sichtbar.

Globale Variable sind im gesamten Programm sichtbar.

Lokale Variable sind in der Regel nur in der Prozedur sichtbar, in der sie verwendet werden.

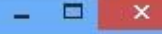
← zurück

+ 14. In prozeduralen Programmen sind in der Regel nicht alle Variable in allen Programmteilen sichtb... ▼

▶ vor

Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- ⊖ Die Frage wurde überprüft und als falsch beantwortet markiert.
- ⊕ Die Frage wurde korrekt beantwortet.
- ℹ Hinweistext



Menu



Aktionen:

Projektübersicht:

aufrufen

aktualisieren

Projekt:

bearbeiten

speichern

download

upload

Programm:

beenden

Frage 15 von 40: Welche Aufgaben hat ein Quelltextanalysator (z.B. Lint) bei der Anwendung auf prozedurale Programme?

- Häufige Missverständnisse von Programmierern aufdecken.
- Syntaxprüfungen durchführen.
- Versuchen, die Absicht des Programmierers zu erkennen.
- Auf Risiken wie Feldüberläufe hinweisen.
- Keine von den übrigen.

zurück

15. Welche Aufgaben hat ein Quelltextanalysator (z.B. Lint) bei der Anwendung auf prozedurale Progr... ▼

vor

Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext

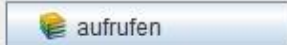


Menu

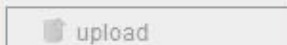


Aktionen:

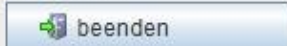
Projektübersicht:



Projekt:

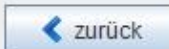


Programm:



Frage 16 von 40: Manche prozeduralen Programmiersprachen enthalten ternäre Operatoren. Welche der folgenden Aussagen treffen zu?

- Ternäre Operatoren liefern drei Werte zurück.
- Ternäre Operatoren liefern genau einen von drei möglichen Werten zurück.
- Keine der übrigen Aussagen trifft zu.
- Ternäre Operatoren haben drei Argumente.
- Ternäre Operatoren haben gegenüber unären und binären stets den höheren Vorrang.

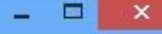


+ 16. Manche prozeduralen Programmiersprachen enthalten ternäre Operatoren. Welche der folgende...



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext

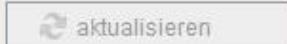
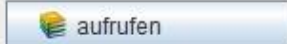


Menu

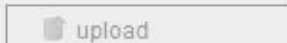
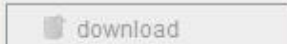


Aktionen:

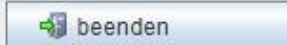
Projektübersicht:



Projekt:

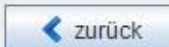


Programm:



Frage 17 von 40: Welche der folgenden Systeme können zur Versionsverwaltung der Quelltexte von prozeduralen Programmen verwendet werden?

- Subversion.
- Revolve.
- Keines von den übrigen.
- CVS.
- Anarchie.

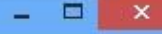


+ 17. Welche der folgenden Systeme können zur Versionsverwaltung der Quelltexte von prozeduralen P...



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext



Menu



Aktionen:

Projektübersicht:

aufrufen

aktualisieren

Projekt:

bearbeiten

speichern

download

upload

Programm:

beenden

Frage 18 von 40: Was gilt bezüglich der Syntax einer prozeduralen Programmiersprache?

- Sie definiert, wie gültige Anweisungen aufgebaut werden.
- Sie legt den Aufbau von gültigen Programmen fest.
- Sie legt fest, aus welchen Konstrukten Programme bestehen können.
- Keine der übrigen Antworten trifft zu.
- Sie umfasst die Lexikalik der Sprache.

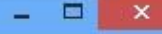
zurück

18. Was gilt bezüglich der Syntax einer prozeduralen Programmiersprache?

vor

Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext

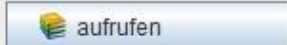


Menu

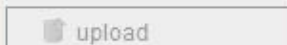


Aktionen:

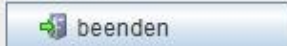
Projektübersicht:



Projekt:

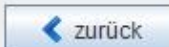


Programm:



Frage 19 von 40: In prozeduralen Sprachen besitzt der Begriff des Sequenzpunktes eine gewisse Bedeutung. Welche Aussagen treffen zu?

- Sequenzpunkte werden genutzt, um Felder mit Werteintervallen zu initialisieren.
- Kommutative arithmetische Operatoren stellen in der Regel keine Sequenzpunkte dar.
- Keine der übrigen Aussagen ist korrekt.
- Am Sequenzpunkt sind alle bisherigen Seiteneffekte evaluiert.
- Der Sequenzpunkt markiert eine Fortsetzungszeile.



+ 19. In prozeduralen Sprachen besitzt der Begriff des Sequenzpunktes eine gewisse Bedeutung. Welc...



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext

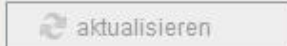
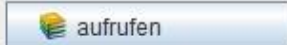


Menu

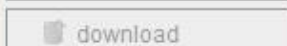


Aktionen:

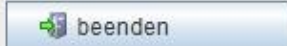
Projektübersicht:



Projekt:

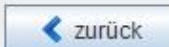


Programm:



Frage 20 von 40: Was gilt bei Funktionsaufrufen in prozeduralen Programmen für Variablenübergabe per Call-by-Reference?

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

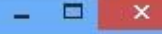


+ 20. Was gilt bei Funktionsaufrufen in prozeduralen Programmen für Variablenübergabe per Call-by-R... ▾



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext

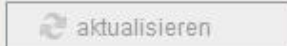
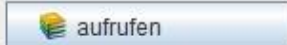


Menu

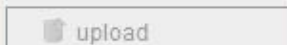


Aktionen:

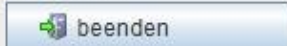
Projektübersicht:



Projekt:

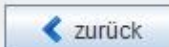


Programm:



Frage 21 von 40: In prozeduralen Programmiersprachen treten Operatoren und Funktionen auf. Durch welche Unterschiede sind sie gekennzeichnet?

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

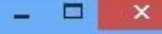


21. In prozeduralen Programmiersprachen treten Operatoren und Funktionen auf. Durch welche Unt...



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext

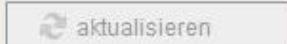
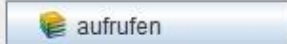


Menu

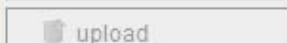


Aktionen:

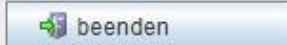
Projektübersicht:



Projekt:

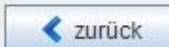


Programm:



Frage 22 von 40: Beim Programmieren in prozeduralen Sprachen kann es geschehen, dass unerreichbarer Code erstellt wird. Welche Aussagen darüber sind richtig?

- Keine der übrigen Aussagen ist richtig.
- Code wird sehr einfach durch unbedingte Sprungbefehle unerreichbar.
- Unerreichbarer Code gelangt niemals zur Ausführung, weil keine Kontrollstruktur den Programmablauf zu ihm führt.
- Viele Compiler erkennen unerreichbaren Code.
- Unerreichbarer Code ist nachteilig hinsichtlich Codeumfang und -wartbarkeit.

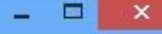


22. Beim Programmieren in prozeduralen Sprachen kann es geschehen, dass unerreichbarer Code ...



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext

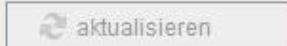
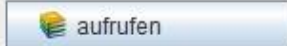


Menu

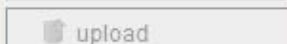


Aktionen:

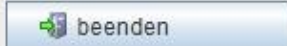
Projektübersicht:



Projekt:

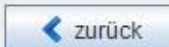


Programm:



Frage 23 von 40: In prozedural erstellten Programmen können Laufzeitfehler auftreten. Viele Programmiersprachen bieten Methoden an, die einen Programmabbruch beim Auftreten solcher Fehler verhindern. Welche Fehler gehören dazu?

- Keiner der übrigen.
- Arithmetische Fehler bei Addition und Subtraktion.
- "Festhängen" in einer Endlosschleife.
- Fließpunktausnahme.
- Unzulässiger Dateizugriff.

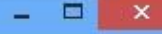


23. In prozedural erstellten Programmen können Laufzeitfehler auftreten. Viele Programmiersprache...



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext

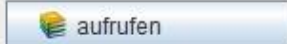


Menu

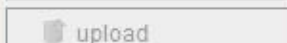


Aktionen:

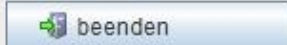
Projektübersicht:



Projekt:

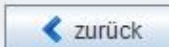


Programm:



Frage 24 von 40: In der prozeduralen Programmierung spielt das Konzept der Rekursion eine bedeutende Rolle. Welche der folgenden Aussagen sind in diesem Zusammenhang zutreffend?

- Jede prozedurale Programmiersprache unterstützt Rekursion.
- Die Prozedurargumente werden beim rekursiven Aufruf in der Regel transparent in einer Stapelstruktur abgelegt.
- Keine der übrigen Aussagen ist zutreffend.
- Rekursive Prozeduren benötigen eine Abbruchbedingung, um endlose Rekursion zu vermeiden.
- Rekursion ist dadurch gekennzeichnet, dass eine Prozedur sich selbst mittelbar oder unmittelbar aufruft.

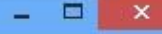


24. In der prozeduralen Programmierung spielt das Konzept der Rekursion eine bedeutende Rolle. ...



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext

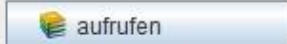


Menu

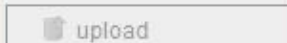
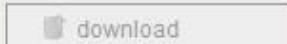


Aktionen:

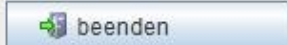
Projektübersicht:



Projekt:

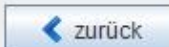


Programm:



Frage 25 von 40: In prozeduralen Programmiersprachen werden Operatoren verwendet, die hinsichtlich ihrer Kardinalität klassifiziert werden können. Welche Aussagen sind zutreffend?

- Die Kardinalität des Additionsoperators ist mindestens zwei.
- Wird für eine unäre und eine binäre Operation dasselbe Symbol verwendet, dann ist die Assoziativität der unären Operation meist höher.
- Die meisten prozeduralen Programmiersprachen besitzen Operatoren mit Kardinalitäten über drei.
- Der Negationsoperator besitzt in der Regel unäre Kardinalität.
- Keine der übrigen Aussagen trifft zu.

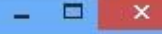


25. In prozeduralen Programmiersprachen werden Operatoren verwendet, die hinsichtlich ihrer Kardi...



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext

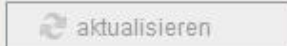
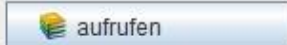


Menu

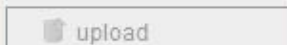


Aktionen:

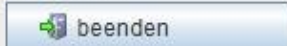
Projektübersicht:



Projekt:

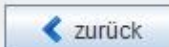


Programm:



Frage 26 von 40: Viele prozeduralen Programmiersprachen unterstützen die Verwendung mehrdimensionaler Felder. In verschiedenen Situationen wirkt sich die Anordnung der Feldelemente im Speicher aus. Welche Aussagen sind in diesem Zusammenhang richtig?

- Keine der übrigen Aussagen ist richtig.
- Die Anordnung der Feldelemente kann sich auswirken, wenn die Felder zur weiteren Verarbeitung gespeichert und an Fremdsysteme übermittelt werden.
- Die Anordnung der Feldelemente ist irrelevant, da im Feld stets Informationen über die Speicherordnung hinterlegt sind.
- Die Anordnung der Feldelemente kann sich auswirken, wenn in einem Programm Prozeduren verschiedener Programmiersprachen zusammengefügt sind.
- In der Regel werden die Feldelemente von der höchsten zur niedrigsten Dimension oder auch von der niedrigsten zur höchsten Dimension abgelegt.



+ 26. Viele prozeduralen Programmiersprachen unterstützen die Verwendung mehrdimensionaler Feld...



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext



Menu



Aktionen:

Projektübersicht:

aufrufen

aktualisieren

Projekt:

bearbeiten

speichern

download

upload

Programm:

beenden

Frage 27 von 40: Zahlreiche prozedurale Programmiersprachen unterstützen Zeiger (Pointer). Welche Aussagen sind in diesem Zusammenhang richtig?

- Ein Zeiger belegt umso mehr Speicher, je größer das Objekt ist, auf das er verweist.
- Sollen Arrays sequentiell verarbeitet werden, dann geschieht dies häufig per Index. Alternativ kann dies durch Zeiger geschehen, die sukzessive auf die Elemente des Feldes verweisen.
- Ein Zeiger enthält die Adresse eines Objektes. Dies kann zum Beispiel die Adresse einer anderen Variablen oder einer Funktion sein.
- Keine der übrigen Aussagen ist richtig.
- Die Typisierung eines Zeigers erfolgt im allgemeinen durch den Datentyp des Objektes, das der Zeiger referenziert.

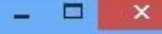
← zurück

+ 27. Zahlreiche prozedurale Programmiersprachen unterstützen Zeiger (Pointer). Welche Aussagen si... ▼

→ vor

Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- ⊖ Die Frage wurde überprüft und als falsch beantwortet markiert.
- ⊕ Die Frage wurde korrekt beantwortet.
- ⓘ Hinweistext

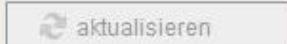
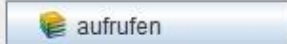


Menu

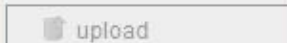
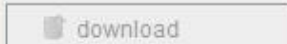


Aktionen:

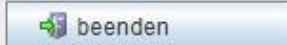
Projektübersicht:



Projekt:

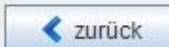


Programm:



Frage 28 von 40: Welche Aufgaben haben Quelltextformatierer (Beautifier wie z.B. indent) bei der Anwendung auf prozedurale Programme?

- Lesbarkeit verbessern
- Einheitliches Layout von Einrückung und Syntaxelementen einhalten.
- Programmierer beim Entwickeln ihres individuellen Programmierstils unterstützen.
- Passende Schriftarten für Schlüsselwörter, Bezeichner usw. wählen.
- Keine von den übrigen.

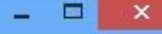


+ 28. Welche Aufgaben haben Quelltextformatierer (Beautifier wie z.B. indent) bei der Anwendung auf pr...



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext

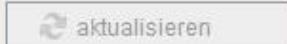
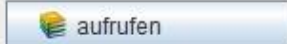


Menu



Aktionen:

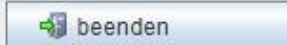
Projektübersicht:



Projekt:

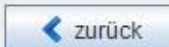


Programm:



Frage 29 von 40: Die meisten prozeduralen Programmiersprachen erlauben die binäre Ein- und Ausgabe von Daten. Was ist zu beachten, wenn diese Daten von Programmen verarbeitet werden sollen, die in verschiedenen Programmiersprachen erstellt sind und möglicherweise auf unterschiedlichen Plattformen laufen?

- Die Typgröße im Sinne des Speicherplatzes ist abhängig von der Programmiersprache.
- Die interne Kodierung binärer Daten (zum Beispiel ihre Endianess) ist von der Plattform abhängig.
- Die Ein- und Ausgabe von Daten im Binärformat erfolgt in der Regel schneller als im Textformat.
- Typgrößen und Kodierung von Binärdaten sind international durch das Völkerrecht festgelegt. Deshalb sollten Daten bevorzugt im Binärformat ein- und ausgegeben werden.
- Keiner der übrigen Sachverhalte ist gesondert zu beachten.

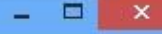


29. Die meisten prozeduralen Programmiersprachen erlauben die binäre Ein- und Ausgabe von Date...



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext

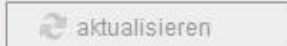
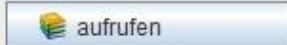


Menu

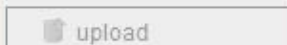


Aktionen:

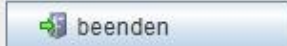
Projektübersicht:



Projekt:



Programm:



Frage 30 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.

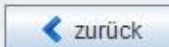
Bei den meisten prozeduralen Programmiersprachen ist es ohne Einschränkung erlaubt, Schlüsselwörter als Bezeichner zu verwenden.

Keine der übrigen Aussagen ist richtig.

Viele prozedurale Programmiersprachen verbieten die Verwendung von Schlüsselwörtern als Bezeichner.

Bei den meisten prozeduralen Programmiersprachen ist es möglich, Schlüsselwörter umzudefinieren und die damit freigewordenen Wörter als Bezeichner zu verwenden.

Im Sprachstandard der jeweilige prozeduralen Programmiersprache ist die Liste der reservierten Schlüsselwörter in der Regel festgelegt.

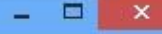


+ 30. Prozedurale Programmiersprachen enthalten normalerweise Schlüsselwörter im definierten Spr...



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- ⊖ Die Frage wurde überprüft und als falsch beantwortet markiert.
- ⊕ Die Frage wurde korrekt beantwortet.
- ⓘ Hinweistext



Menu

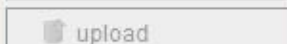


Aktionen:

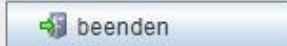
Projektübersicht:



Projekt:

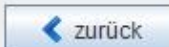


Programm:



Frage 31 von 40: Was gilt in den meisten prozeduralen Sprachen für die Benennung von Bezeichnern?

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

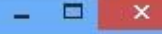


+ 31. Was gilt in den meisten prozeduralen Sprachen für die Benennung von Bezeichnern?



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext

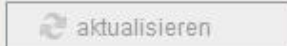
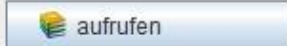


Menu

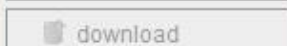


Aktionen:

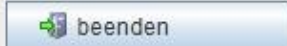
Projektübersicht:



Projekt:



Programm:



Frage 32 von 40: Welches ist die richtige Reihenfolge der Buchstaben des Wortes "COMPILER" im ASCII-Code?

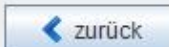
83, 67, 69, 81, 72, 79, 65, 88.

65, 66, 67, 68, 69, 70, 71, 72.

Keine von den übrigen.

67, 79, 77, 80, 73, 76, 69, 82.

67, 69, 73, 76, 77, 79, 80, 82.

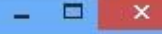


+ 32. Welches ist die richtige Reihenfolge der Buchstaben des Wortes "COMPILER" im ASCII-Code?



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- ⊖ Die Frage wurde überprüft und als falsch beantwortet markiert.
- ⊕ Die Frage wurde korrekt beantwortet.
- ⓘ Hinweistext

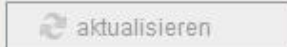


Menu

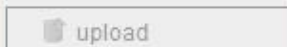
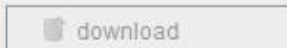


Aktionen:

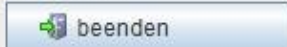
Projektübersicht:



Projekt:

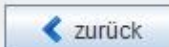


Programm:



Frage 33 von 40: Datenstrukturen - welche Aussagen treffen zu?

- Keine von den übrigen.
- 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.
- Häufig verwendete Datenstrukturen sind Felder oder Arrays.
- Komplexere Datenstrukturen werden aus elementaren Typen aufgebaut.

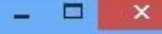


+ 33. Datenstrukturen - welche Aussagen treffen zu?



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext

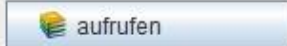


Menu

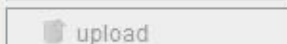


Aktionen:

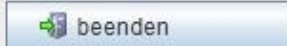
Projektübersicht:



Projekt:

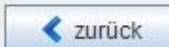


Programm:



Frage 34 von 40: Welche Aufgaben hat ein Projektmanagementwerkzeug (z.B. Make)?

- Erzeugen des Softwareproduktes (z.B. Programm) aus seinen Quellen.
- Keine von den übrigen.
- Arbeitszeitmanagement der Programmierer.
- Koordination des Informationsflusses im Entwicklerteam.
- Organisation von Compiler- und Linkerläufen automatisieren.

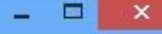


34. Welche Aufgaben hat ein Projektmanagementwerkzeug (z.B. Make)?



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext

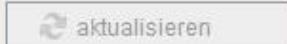
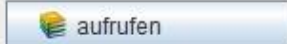


Menu



Aktionen:

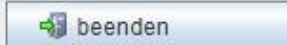
Projektübersicht:



Projekt:

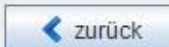


Programm:



Frage 35 von 40: Was trifft bezüglich der Initialisierung von Variablen in prozeduralen Programmiersprachen zu?

- In jeder prozeduralen Programmiersprache werden Variablen auch ohne explizite Initialisierung mit Null vorbelegt.
- Keine der übrigen Aussagen ist korrekt.
- Initialisierte statische Variable werden nur einmal initialisiert.
- Variableninitialisierung wirkt sich auf die Größe der compilierten Binärdatei aus.
- Initialisierte dynamische Variable werden bei jedem Aufruf der Prozedur, in der sie deklariert sind, initialisiert.

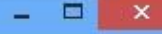


35. Was trifft bezüglich der Initialisierung von Variablen in prozeduralen Programmiersprachen zu?



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext



Menu



Aktionen:

Projektübersicht:

aufrufen

aktualisieren

Projekt:

bearbeiten

speichern

download

upload

Programm:

beenden

Frage 36 von 40: Bei der Notation prozeduraler Programme sind Regeln und Konventionen zu beobachten. Welche Aussagen treffen zu?

- Viele Programmiersprachen bieten ein Symbol an, das einen Kommentar bis zum Zeilenende einleitet.
- Einige Programmiersprachen verlangen explizit ein Symbol zum Abschluss eines Statements; dieses darf sich dann über mehrere Zeilen erstrecken.
- Anfang und Ende von Blöcken werden häufig durch spezielle Klammersymbole oder durch entsprechende Schlüsselwörter gekennzeichnet.
- Die Blöcke in Schleifen und Fallunterscheidungen werden meist durch Einrückung kenntlich gemacht.
- Keine der übrigen Aussagen trifft zu.

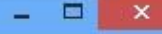
zurück

36. Bei der Notation prozeduraler Programme sind Regeln und Konventionen zu beobachten. Welche... ▼

vor

Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext



Menu

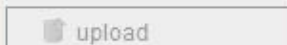


Aktionen:

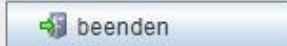
Projektübersicht:



Projekt:

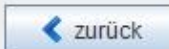


Programm:



Frage 37 von 40: In der prozeduralen Softwareentwicklung spielt das Paradigma der strukturierten Programmierung eine bedeutende Rolle. Welche Aussagen sind in diesem Zusammenhang zutreffend?

- Die strukturierte Programmierung geht unter anderem auf Arbeiten von Edsger Dijkstra zurück.
- Das Konzept der strukturierten Programmierung schreibt vor, dass Prozeduren in den Quelltexten in ihrer alphabetischen Reihenfolge angeordnet werden.
- Keine der übrigen Aussagen ist zutreffend.
- Beim Konzept der strukturierten Programmierung wird auf unbedingte Sprunganweisungen möglichst verzichtet.
- Wiederverwendbare Teilsequenzen werden bei der strukturierten Programmierung möglichst in Prozeduren zusammengefasst.

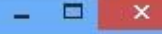


+ 37. In der prozeduralen Softwareentwicklung spielt das Paradigma der strukturierten Programmierun...



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext

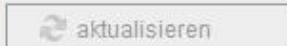
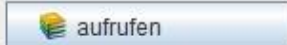


Menu

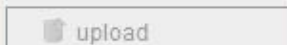


Aktionen:

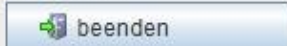
Projektübersicht:



Projekt:

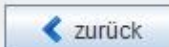


Programm:



Frage 38 von 40: In den meisten prozeduralen Programmiersprachen werden unterschiedliche Arten von Schleifen unterstützt. Welche gehören dazu?

- Abweisende Schleifen.
- Zuweisende Schleifen.
- Anweisende Schleifen.
- Vorweisende Schleifen.
- Keine der übrigen.

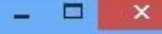


+ 38. In den meisten prozeduralen Programmiersprachen werden unterschiedliche Arten von Schleifen...



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext

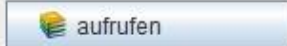


Menu



Aktionen:

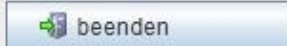
Projektübersicht:



Projekt:

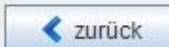


Programm:



Frage 39 von 40: Ein System, auf dem prozedural programmierter Code ausgeführt werden soll, muss bestimmte Eigenschaften aufweisen. Welche gehören dazu?

- Das System muss eine CPU besitzen.
- Das System muss keine der übrigen Eigenschaften aufweisen.
- Das System muss virtuelle Speicherverwaltung unterstützen.
- Auf dem System muss ein Compiler verfügbar sein.
- Das System muss genügend RAM besitzen, um den Programmcode darin abzulegen.

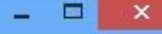


39. Ein System, auf dem prozedural programmierter Code ausgeführt werden soll, muss bestimmte ...



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext

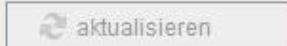
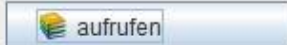


Menu



Aktionen:

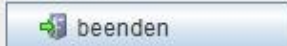
Projektübersicht:



Projekt:

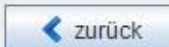


Programm:



Frage 40 von 40: Ein wichtiges Instrument bei der Qualitätssicherung prozeduraler Programme ist der Debugger. Welche Aussagen über seine Funktionen sind zutreffend?

- Mit einem Debugger können zur Laufzeit Variableninhalte verändert werden.
- Keine der übrigen Aussagen ist richtig.
- Ein Debugger kann verwendet werden, um zur Laufzeit die Reihenfolge von Funktionsaufrufen zu ermitteln.
- Der Debugger kann zur Laufzeit an bestimmten Verarbeitungspunkten - sogenannten Breakpoints - die Ausführung stoppen, um beispielsweise Variableninhalte zu präsentieren.
- Der Debugger kann zur Laufzeit die Verarbeitungsfolge aufzeigen.



40. Ein wichtiges Instrument bei der Qualitätssicherung prozeduraler Programme ist der Debugger. ...



Auswahlbox Legende

- Die Frage wurde noch nicht bearbeitet.
- Die Frage wurde beantwortet aber noch nicht überprüft.
- Die Frage wurde überprüft und als falsch beantwortet markiert.
- Die Frage wurde korrekt beantwortet.
- Hinweistext