Exámenes

Self-Assessment Test Theme 2

Volver a la Lista de Examenes

Parte 1 de 3 - Second 0.67/ 2.0 Puntos

Preguntas 1 de 10

1.0/ 1.0 Puntos. Puntos descontados por fallo: 0.33

Consider the following definition of an *identifier* using BNF notation

```
<digito>::= <par>|<impar>
<par> ::= 0|2|4|6|8
<impar>::= 1|3|5|7|9
<letra>::= x<letra>|y<par>|z<impar>
<identificador>::=<letra><digito>
```

Which of the following identifiers is **NOT** legal in the language defined by the previous rules?

- A. `xxy00`
- B. `xxy01`
- C. `z00`
- D. `z11`

Preguntas 2 de 10

-0.33/ 1.0 Puntos. Puntos descontados por fallo: 0.33

The following declaration of variables of a program in C

```
itn a
```

contains an error produced by misspelling the name of the default type for integers. This error is detected by the

- A. scanner
- B. parser
- C. semantic analyzer
- D. linker

Parte 2 de 3 - First 3.5/ 3.5 Puntos

Preguntas 3 de 10

0.5/ 0.5 Puntos. Puntos descontados por fallo: 0.33

Two programs **P1** and **P2** are *equivalent* (**P1 ≈ P2**) if they have the same semantics. Considering the *big step* operational semantics as a basis to define such an equivalence, which of the following equivalence statements is **TRUE**?

- A. (x:= 5; while x>2 do x:= x-1) ≈ (x:= 1; x:= x+x)
- B. (P1 ≈ P2), siendo

  P1: x := 0; y := 2;
  if false then x:= y else y:= x;
  x := y;

  P2: x := 0; y := 2;
  if false then x := y else {y := x; x := y};

- C. (x := 1; while x < 2 do x:= x + 3) ≈ (x := 1)
- D. (x:=1; y:=3; if x <= y then x:=0 else x:=1) ≈ (x:=1; y:=3)

Preguntas 4 de 10

0.5/ 0.5 Puntos. Puntos descontados por fallo: 0.33

Complete the following sentence:

"The following axiomatic semantic rule:
wp(X := exp;Q) = Q[X ↦ exp]

establishes that the weakest precondition of the assignment instruction X:=exp with respect to the postcondition is obtained by ..."

- A. replacing exp in Q by X.
- B. adding the expression X ↦ exp to Q.
- C. removing the expression X ↦ exp from Q.
- D. replacing every occurrence of X in Q by exp.

Preguntas 5 de 10

0.5/ 0.5 Puntos. Puntos descontados por fallo: 0.33

Which configuration is required (in *) to make the following evaluation complete by using the small-step operational semantics?

⟨if X>Y then Y:=Y+X else Y:=0, {X ↦→ 42, Y ↦ 0}⟩
⟨X>Y, {X ↦ 42, Y ↦ 0}⟩ ⇒ true
⟨X, {X ↦ 42, Y ↦ 0}⟩ ⇒ 42
⟨Y, {X ↦ 42, Y ↦ 0}⟩ ⇒ 0
→ ⟨Y:=Y+X, {X ↦ 42, Y ↦ 0}⟩
⟨Y+X, {X ↦ 42, Y ↦ 0}⟩ ⇒ 42
⟨Y, {X ↦ 42, Y ↦ 0}⟩ ⇒ 0
⟨X, {X ↦ 42, Y ↦ 0}⟩ ⇒ 42
→ (*

- A. ⟨ Y:=0, {X ↦ 42, Y ↦ 0} ⟩
- B. ⟨ skip, {X↦ 42, Y ↦ 0} ⟩
- C. ⟨ if X>Y then Y:=Y+X else Y:=0, {X ↦ 42, Y ↦ 0} ⟩
- D. ⟨ skip, {X ↦ 42, Y ↦ 42} ⟩

Preguntas 6 de 10

0.5/ 0.5 Puntos. Puntos descontados por fallo: 0.33

Consider the following transition rules (for assignment and sequencing) of the small- step operational semantics of the small imperative language IMP studied in the course

$$\frac{\langle a, e\rangle \Rightarrow n}{\langle X := a, e\rangle \rightarrow \langle skip, e[X \mapsto n]\rangle}$$

$$\frac{\langle i0, e\rangle \rightarrow \langle i'0, e'\rangle}{\langle (i0; i1), e\rangle \rightarrow \langle (i'0; i1), e'\rangle} \qquad \frac{}{\langle (skip; i), e\rangle \rightarrow \langle i, e'\rangle}$$

Fill the gap below for an appropriate continuation of the following execution trace:

$\langle x:=x+y;y:=x+1,\{x\rightarrow 1,y\rightarrow 1\}\rangle$

$\langle x:=x+y,\{x\rightarrow 1,y\rightarrow 1\}\rangle$

$\langle x+y,\{x\rightarrow 1,y\rightarrow 1\}\rangle \Rightarrow 2$

$\rightarrow \langle skip,\{x\rightarrow 2,y\rightarrow 1\}\rangle$

$\rightarrow \langle skip;y:=x+1,\{x\rightarrow 2,y\rightarrow 1\}\rangle$

$\rightarrow \langle y:=x+1,\{x\rightarrow 2,y\rightarrow 1\}\rangle$

?

- A. $\rightarrow \langle skip,\{x\mapsto 2,y\mapsto 1\}\rangle$
- B. $\langle x+1,\{x\mapsto 2, y\mapsto 1\}\rangle \Rightarrow 3$
  $\rightarrow \langle skip,\{x\mapsto 2, y\mapsto 3\}\rangle$
- C. $\{x\mapsto 2, y\mapsto 1\} \Rightarrow 3$
  $\rightarrow \langle \{x\mapsto 2, y\mapsto 3\}\rangle$
- D. $\langle x+1,\{x\mapsto 2,y\mapsto 1\}\rangle \Rightarrow 3$

Preguntas 7 de 10

0.5/ 0.5 Puntos. Puntos descontados por fallo: 0.33

The big-step operational semantics

- A.
  establishes a relationship between the initial configuration $\langle P,e_0\rangle$ of the program and the final state e_f after the execution.
- B.
  yields the sequence of configurations (*trace*) which are obtained during the step-by-step execution of the program.
- C. can be used to establish the correctness of the program by means of the weakest precondition calculus.
- D. cannot be used to guide the implementation of programming languages.

Preguntas 8 de 10

0.5/ 0.5 Puntos. Puntos descontados por fallo: 0.33

Given a semantics S for a programming language, we say that two programs **i1** and **i2** are equivalent (written **i1 ≈ i2**) if they have the same semantics. If we consider the *big step* operational semantics for SIMP, which of the following program equivalence statements is **WRONG**?

- A. (y:=y) ≈ skip

  (donde skip es la instrucción vacía)
- B. (x:=y; y:=x) ≈ (y:=x; x:=y)
- C. (x:=y; y:=x) ≈ x:=y
- D. (if x >= 0 then x:= x-x else x:=x*0) ≈ x:=0

Preguntas 9 de 10

0.5/ 0.5 Puntos. Puntos descontados por fallo: 0.33

Supposing that we extend the syntax of SIMP with the instruction =:= and add the following small-step semantic rule:

$$\frac{\langle X,e\rangle \Rightarrow n_0 \qquad \langle Y,e\rangle \Rightarrow n_1}{\langle X =:= Y, e\rangle \rightarrow \langle skip, e[X \mapsto n_1][Y \mapsto n_0]\rangle}$$

Which of the following sentences defines its meaning:

- A. It's a regular assignment but restricted to two variables in such a way that X is assigned the value of Y.
- B. Both variables X and Y are assigned an initial value.
- C. It exchanges the values of variables X and Y.
- D. Checks whether the values of variables X and Y are the same or not.

Parte 3 de 3 - Third 1.0/ 1.0 Puntos

Preguntas 10 de 10

1.0/ 1.0 Puntos. Puntos descontados por fallo: 0.33

Which of the following sentences is **TRUE**

- A.
  In pure compiled languages each instruction is simultaneously analyzed and executed, so that programs are more efficient.
- B. Pure interpreted languages are high-level languages which are translated to intermediate code.
- C.
  In programming languages with a mixed implementation scheme the original language is translated into an intermediate language which is then interpreted.
- D.
  In pure programming languages the original language is translated into an object language which is then interpreted.