

TERCER PARCIAL EXAMEN DE LÓGICA

Trimestre 12-P. Junio 26 de 2012.

Grupo: CCB02 **Profesor:** Dr. Carlos Barrón Romero

Matricula: _____

Nombre del **Alumno:** _____

Instrucciones. Conteste todas las preguntas.

El marco de sus respuestas y comprensión de los temas de Lógica son los objetivos de la UEA de Lógica (clave 111222) que transcribo a continuación:

1. Comprender los principios básicos de la lógica matemática.
2. Demostrar la validez de argumentos mediante reglas formales.
3. Aplicar principios de lógica matemática en la elaboración de programas de cómputo.

TEORIA

Los esquemas clásicos de inferencia.

Modus Ponendo Ponens $\frac{p \Rightarrow q}{q}$. Doble negación $\frac{p}{\neg\neg p}$.

Modus Tollendo Tollens $\frac{p \Rightarrow q}{\neg q}$. Regla de Adjunción $\frac{p}{p \wedge q}$.

Modus Tollendo Ponens $\frac{p \vee q}{q}$ $\frac{p \vee q}{p}$.

Demostrar (es decir dar o nombrar los esquemas o las equivalencias o el álgebra para obtener la proposición que se pide demostrar o explicar que no se puede porque se tiene un error).

Punto flotante y aritmética(PF): significa que solo se pueden representar valores entre $[-1 \times 10^{36}, 1 \times 10^{36}]$ y el número más pequeño es 1×10^{-36} y la mantisa en esta notación de punto flotante es de siete dígitos. La conversión se hace flotando el valor y recortándolo a 7 dígitos de mantisa. Un overflow ocurre cuando una operación sale del rango de valores.

PREGUNTAS

1) **(20)** Explicar los detalles (que regla o equivalencia u operación algebríaca se usa) y justificar si es correcta o incorrecta toda la argumentación de la inferencia de la demostración siguiente.

Demostrar $p(5)$, dado $(\forall x, x \in \mathbb{N}, q(x) \Rightarrow (p(x) \vee r(x)))$, $q(5)$, $\neg\neg\neg r(5) \wedge 1$.

Demostración.

$(\forall x, x \in \mathbb{N}, q(x) \Rightarrow (p(x) \vee r(x))) \Rightarrow 5 \in \mathbb{N}, q(5) \Rightarrow (p(5) \vee r(5))$,

$q(5) \Rightarrow (p(5) \vee r(5))$

$\frac{q(5)}{p(5) \vee r(5)}$,

$\neg\neg\neg r(5) \wedge 1 \equiv \neg\neg\neg r(5)$,

$\neg\neg\neg r(5) \equiv \neg r(5)$,

$p(5) \vee r(5)$

$\frac{\neg r(5)}{p(5)}$.

2) **(20)** Traduzca a predicados o enunciados y justifique su definición de equivalencia de la compilación de programas del siguiente predicado:

Todos los programas de cómputo de un lenguaje de alto nivel que se compilan, generan un programa compilado con un código de instrucciones y declaraciones de un lenguaje de máquina equivalente.

Sugerencia, ajuste el predicado: $\forall D, p \text{ y } m \text{ son equivalentes} \Rightarrow p(D)=m(D)$.

Donde p es un programa en lenguaje de alto nivel, m es un programa en lenguaje de máquina, D es un conjunto de datos de entrada, $p(D)=\{\text{resultados que genera } p \text{ con } D\}$ y $m(D)=\{\text{resultados que genera } m \text{ con } D\}$.

3) **(20)** Justifique que predicados o enunciados (a, b o c) se obtienen lógicamente de:

Sea p , el programa de computo, que con el conjunto de datos D , funciona y en un tiempo finito crea el conjunto de resultados R .

a) $\forall E$ (conjunto de datos de entrada adecuados), $E \neq D \Rightarrow p$ funciona y en tiempo finito crea el conjunto de resultados correspondiente a E .

b) $p(D) \Rightarrow R$.

c) $p(D) = R$.

4) Dado el siguiente programa en lenguaje C y PF con el esquema de correctez, explique y justifique sus respuestas.

a) **(10)** De un ejemplo de un valor de a , donde termina en tres iteraciones y escriba los resultados de las variables c e i .

b) **(10)** De un par de valores distintos de a , donde no termina y explique porque no termina.

c) **(10)** De un ejemplo de un valor de a , donde termina y se obtenga un resultado exacto en la variable c .

d) **(10)** De un ejemplo de un valor de a , donde termina y se obtenga un resultado no exacto en la variable c .

```
#include <iostream>
int main() {
    int i=0;
    float c=1.0;
    float a;
    printf("\nDar un valor positivo para a => ");
    scanf("%f",&a);
    while (1) {
        i=i+1;
        c=c+a;
        if (c>3.0)
            break;
    }
    printf("\nEl valor de c=\%f, iteraciones=%d\n",c,i);
    system("PAUSE");
    return EXIT_SUCCESS;
}
```