

Proyecto-ExamenGrupo1-IEEE754

December 2, 2024

Proyecto Examen Bi-Mestral de Arquitectura de computadores

Integrantes:

- Lenin Amangandi
- Richard Tipantiza

Fecha de presentacion: 2024-12-02

0.1 Objetivo

Realizar un programa en el lenguaje de programación Python que permita a un número decimal representarlo en formato IEEE de 32 bits.

0.2 Función que permita convertir un número decimal a su representación binario.

La función `convertirABinario` consiste en tener como parámetro el número decimal y dividirlo secuencialmente hasta que se obtenga un residuo de 0 lo que nos lleva a utilizar un ciclo `while`.

Adicionalmente utilizamos un casteo para convertir el tipo de dato numérico a string como se muestra en el siguiente bloque de código.

```
[20]: def convertirABinario(parteEntera):  
    if parteEntera == 0:  
        return '0'  
  
    resultadoBinario = ''  
    while parteEntera > 0:  
        resultadoBinario = str(parteEntera % 2) + resultadoBinario  
        parteEntera //= 2  
    return resultadoBinario
```

0.3 Función que permita convertir la parte decimal de un número a su representación binario.

Se toma un número decimal como entrada, multiplicándolo repetidamente por 2, extrayendo el dígito entero (0 o 1) y agregándolo a una lista. Finalmente, devuelve la lista convertida a una cadena de texto, que es el equivalente binario de la parte fraccionaria original.

```
[21]: def fraccionABinario(parteFraccionaria):
    if parteFraccionaria == 0:
        return ''
    resultadoBinario = []
    while parteFraccionaria != 1:
        parteFraccionaria *= 2
        digitoBinario = int(parteFraccionaria)
        resultadoBinario = resultadoBinario + [str(digitoBinario)]
        parteFraccionaria -= digitoBinario
        if len(resultadoBinario) > 23:
            break
    return ''.join(resultadoBinario)
```

0.4 Función convertir de decimal a su representación IEEE.

Se toma un número decimal como entrada, separando la parte entera y la parte fraccionaria, convirtiéndolas a binario, calculando el exponente y la mantisa, y finalmente creando la representación de 32 bits en formato IEEE 754.

```
[19]: def convetirIEEE32bits(numero):
    signoBit = 0 if numero >= 0 else 1
    if numero < 0:
        numero = -numero

    parteEntera = int(numero)
    parteFraccionaria = numero - parteEntera

    enteroBinario = conversionBinario(parteEntera)
    fraccionariaBinario = fraccionABinario(parteFraccionaria)

    if parteEntera != 0:
        exponente = len(enteroBinario) - 1
        mantisa = enteroBinario[1:] + fraccionariaBinario
    else:
        exponente = -fraccionariaBinario.find('1') - 1
        mantisa = fraccionariaBinario[fraccionariaBinario.find('1') + 1:]

    exponenteSesgado = exponente + 127

    mantisa = mantisa[:23]
    mantisa = mantisa + '0' * (23 - len(mantisa))
    binarioIEEE32bits = f'{signoBit:1b}{exponenteSesgado:08b}{mantisa}'
    hexadecimal = hex(int(binarioIEEE32bits, 2))
    return {
        "numero": numero,
        "signoBit": signoBit,
        "exponente": exponente,
```

```

        "exponenteSesgado": exponenteSesgado,
        "mantisa": mantisa,
        "binarioIEEE32bits": binarioIEEE32bits,
        "hexadecimal": hexadecimal
    }

```

```

[18]: def imprimir_datos_ieee32bits(resultado):

    print("Número dado:", resultado["numero"])
    print("Bit del signo:", resultado["signoBit"])
    print("Exponente:", resultado["exponente"])
    print("Exponente sesgado:", resultado["exponenteSesgado"])
    print("Mantisa:", resultado["mantisa"])
    print("Binario en formato IEEE:", resultado["binarioIEEE32bits"])
    print("Equivalente hexadecimal:", resultado["hexadecimal"])

```

0.5 Ejecución de la calculadora desde la clase inicial

1. Pide al usuario que ingrese un número decimal.
2. Llama a la función `convetirIEEE32bits` para convertir el número decimal a su representación en formato IEEE 754 de 32 bits.
3. Imprime la información sobre la representación en formato **IEEE 754 de 32 bits**, incluyendo:
 - El bit de signo
 - El exponente
 - El exponente sesgado
 - La mantisa
 - La representación binaria
 - La representación hexadecimal.

```

[17]: numero = float(input("Introduce un número decimal: "))
    resultado = convetirIEEE32bits(numero)
    imprimir_datos_ieee32bits(resultado)

```

Introduce un número decimal: 47

Número dado: 47.0

Bit del signo: 0

Exponente: 5

Exponente sesgado: 132

Mantisa: 011110000000000000000000

Binario en formato IEEE: 01000010001111000000000000000000

Equivalente hexadecimal: 0x423c0000