

INFORME III

“Conversión de números decimales a otras bases”

Como grupo hemos parametrizado la función que logra el cambio de base ya que, básicamente, siempre debemos repetir los mismos pasos.

Pasos a seguir:

1. Número decimal se debe dividir por la base elegida
2. Se debe guardar el módulo en un arreglo
3. Se iteran estos pasos hasta que el resultado de la división anterior de 1.

Cabe destacar que el código desarrollado en C contempla solo números enteros y el código Python contempla, además de los enteros, los decimales fraccionarios.

En necesario, hacer un cambio para la transformación hexadecimal, por lo mismo, si el módulo es mayor o igual a 10, se deben tener en cuenta los siguientes pasos:

1. Si el módulo es 10, se guarda la letra 'A' en el arreglo de módulos.
2. Si el módulo es 11, se guarda la letra 'B' en el arreglo de módulos.
3. Si el módulo es 12, se guarda la letra 'C' en el arreglo de módulos.
4. Si el módulo es 13, se guarda la letra 'D' en el arreglo de módulos.
5. Si el módulo es 14, se guarda la letra 'E' en el arreglo de módulos.
6. Si el módulo es 15, se guarda la letra 'F' en el arreglo de módulos.

El arreglo de módulos debe ser invertido para obtener el resultado correcto.

Hemos implementado dos soluciones para esta tarea, en Python y en C.

Pseudocódigo Parametrizado

```
entero numero, resto[100];  
leer(numero)
```

```
desde(entero i=0; numero>1; i++){  
    resto[i]=numero%base;  
    numero=numero/base;  
}
```

```
resto[i]=numero;
```

```

desde(i;i>=0;i--)
imprimir(resto[i]);
}

```

Donde numero es el numero decimal

El vector resto es la secuencia de números convertidos

Pseudocódigo parte Decimal

parte_decimal(valor, base):

```

    num_dec=valor - par_entera_valor
    counter=0
    while (num_dec-entero(num_dec))!=0:
        counter+=1
        num_dec=num_dec*10
        num_dec=valor - par_entera_valor
    for i in range(counter to 0):
        num_dec=num_dec*base
        valor = parte_entera_num_dec
        if valor>=10:
            valor = Representación es ascii desde 'A' hasta 'Z'
        num_final+=string(valor)
        num_dec=num_dec-parte_entera_num_dec
    retorno num_final

```

- **Enlace de Github:** <https://github.com/darkmeow/arqui>