

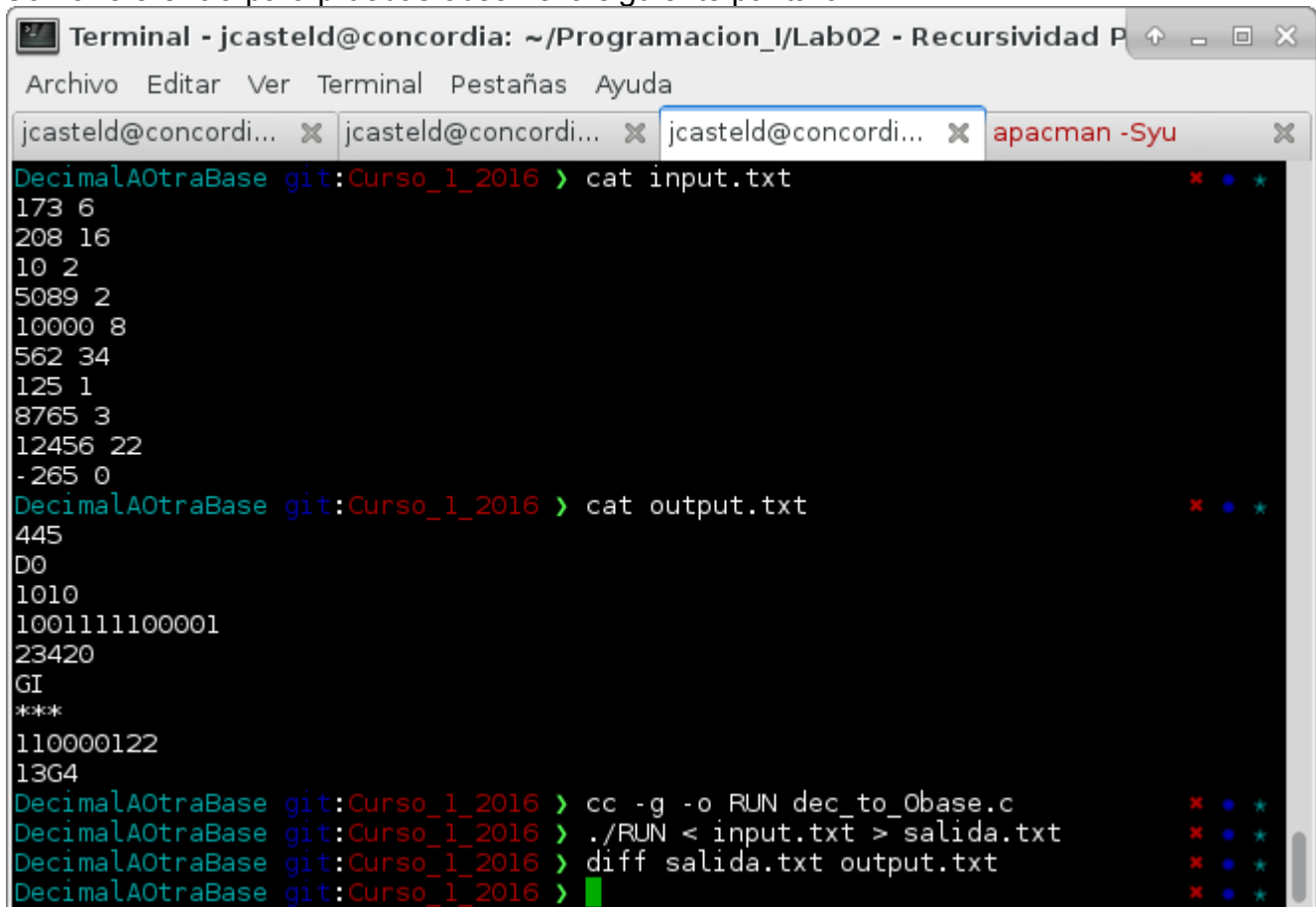
## Cambio de base 10 a otra base n

Diseñar y codificar en C un programa recursivo que nos permita cambiar de base, el cambio sera de la base 10 a otra base b, en este caso la base b estará entre 2 y 36, ademas el número a cambiar de base estará en el rango [1, 50000]

Para los dígitos mayores que 9 puede usar el alfabeto de mayúsculas; así su sistema de numeración base va a ser: '0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ'.

### Casos de prueba:

Como referencia para pruebas observe la siguiente pantalla:



```
Terminal - jcasteld@concordia: ~/Programacion_I/Lab02 - Recursividad P
Archivo Editar Ver Terminal Pestañas Ayuda
jcasteld@concordi... x jcasteld@concordi... x jcasteld@concordi... x apacman -Syu x
DecimalA0traBase git:Curso_1_2016 > cat input.txt
173 6
208 16
10 2
5089 2
10000 8
562 34
125 1
8765 3
12456 22
-265 0
DecimalA0traBase git:Curso_1_2016 > cat output.txt
445
D0
1010
1001111100001
23420
GI
***
110000122
13G4
DecimalA0traBase git:Curso_1_2016 > cc -g -o RUN dec_to_Obase.c
DecimalA0traBase git:Curso_1_2016 > ./RUN < input.txt > salida.txt
DecimalA0traBase git:Curso_1_2016 > diff salida.txt output.txt
DecimalA0traBase git:Curso_1_2016 >
```

Cada uno de los casos de prueba representa una linea en el archivo de entrada. Cada linea a su vez contiene dos números. El primer **número** es el decimal que se desea convertir y el segundo es la **base** a la cual se desea convertir.

Para la primera linea el **número** = 173 y la **base** = 6

Para la segunda linea el **número** = 208 y la **base** = 16

En la séptima linea se observa el **número** = 125 pero la **base** = 1 → la salida es \*\*\*

La última linea contiene un **número negativo** y entonces se considera que finalizan las pruebas.

Tenga en cuenta que para sus pruebas debería elaborar un archivo de personalizado de

pruebas, ya que la prueba final se realizará con un archivo generado en forma aleatoria. No olvide validar los casos extremos: base mínima, base máxima, número mínimo y número máximo. El tamaño máximo del string de salida depende del número máximo a convertir y de la base utilizada.

**Sugerencias:**

Para obtener una estrategia para diseñar su algoritmo recursivo de cambio de base puede ver este enlace:

<http://www.hitmontop-ejerciciosresueltos.blogspot.mx/2013/05/de-decimal-binario-con-una-funcion.html>

Dependiendo de su estrategia para resolver este ejercicio puede necesitar revisar el contenido de strings en C.

Libro Joyanes Aguilar. Programación en C.

[http://courses.cs.vt.edu/cs2505/spring2014/Notes/T22\\_CStrings.pdf](http://courses.cs.vt.edu/cs2505/spring2014/Notes/T22_CStrings.pdf)

[http://www.tutorialspoint.com/cprogramming/cprogramming\\_tutorial.pdf](http://www.tutorialspoint.com/cprogramming/cprogramming_tutorial.pdf)

Se sugiere revisar las funciones: `strlen` `strcpy` `strcat`.

<http://c.conclase.net/librerias/index.php?ansilib=string#inicio>

Evitar el uso de la función `gets`.

<http://stackoverflow.com/questions/1694036/why-is-the-gets-function-so-dangerous-that-it-should-not-be-used>

Es probable que necesite el uso de las funciones `malloc` y/o `calloc` para reservar memoria para un string. La función `free` para liberar memoria.

[http://www.tutorialspoint.com/c\\_standard\\_library/c\\_function\\_malloc.htm](http://www.tutorialspoint.com/c_standard_library/c_function_malloc.htm)