66.20 Organización de Computadoras
Trabajo Práctico #0:
Infraestructura básica

97043 - González Pérez, Ailén Y.
96260 - Mariotti, María Eugenia
95457 - Raña, Cristian Ezequiel

4 de Abril de 2017

1. Diseño e implementación

1.1. Diseño del programa

El programa se divide en dos funciones principales: codificador y decodificador. El codificador transforma expresiones con caracteres ASCII en base64, mientras que el decodificador hace el proceso inverso. Ambas funciones se implementan con diversos ciclos y recorridos, así como utilización de instrucciones aritmético lógicas (sumas, , shifts, etc) que relacionan la codificación base 64 con ASCII.

A su vez, se cuenta con una función readerçuyo objetivo es obtener, de la entrada, un string que sirva como parámetro para encode y decode.

Se definen las funciones auxiliares show_help y show_version, las cuales tienen como objetivo clarificar el código en main, evitando poner en esta función todas las líneas que componen estas funciones.

Se diseñan tres archivo de pruebas. Esta decisión tiene como objetivo probar de forma clara y rápidamente legible diversos resultados de nuestro programa, agrupando de acuerdo al tipo de prueba.

1.2. Parámetros

Se detallan a continuación los parámetros del programa

- -h: Visualiza la ayuda del programa, en la que se indican los parámetros y sus objetivos.
- -V: Indica la versión del programa.
- -i: Archivo de entrada del programa.
- -o: Archivo de salida del programa.
- -a: Acción a llevar a cabo: codificación o decodificación.

Se indica a continuación detalles respecto a los parámetros:

- Si no se explicitan -i y -o, se utilizarán stdin y stdout, respectivamente. Si no se explicita -a, se realizará una codificación, ya que encode es la opción por defecto
- -V es una opción ”show and quit”. Si se explicita este parámetro, sólo se imprimirá la versión, aunque el resto de los parámetros se hayan explicitado. La única excepción es la ayuda (-h); si este parámetro se especifica antes que -V, primero se imprimirá la ayuda y luego se procederá al ”show and quit”version.
- -h también es de tipo ”show and quitz se comporta de forma similar a -V.
- en caso de que se use la entrada estándar (con comando echo texto — ./tp0 etc) y luego se especifique un archivo de salida con -i, prevalecerá el establecido por parámetro.

2. Documentación del proceso de compilación y ejecución

2.1. Compilación y ejecución

La compilación del archivo fuente se realiza de la siguiente manera

```
$ gcc -std=c99 -g -Wall main.c -o tp0
```

Para proceder a la ejecución del programa, se debe llamar a (según el nombre elegido al compilar):

```
$ ./tp0
```

seguido de los parámetros que se desee modificar, los cuales se indicaron en la sección 1.2.
En caso de ser entrada estándar (stdin) se podrá ejecutar de la siguiente forma:

```
$ echo textoDeEntrada | ./tp0
```

También en este caso, se indican a continuación los parámetros a usar.
Se puede repetir el proceso para ejecutar en netBSD.

3. Corridas de prueba

Todas las pruebas que se presentan a continuación, están codificadas en los archivos de prueba ***.txt de forma que puedan ejecutarse y comprobar los resultados obtenidos.[1]

Presentaremos: identificación de las pruebas, comandos para ejecutarlas, líneas de código que las componen y resultado esperado.

Antes de todas las ejecuciones de pruebas, debe ejecutarse el comando

```
$gcc -std=c99 -g -Wall -o tp0 *c
```

3.0.1. Generales

Todas se van a ejecutar cuando hagamos

```
$chmod +x pruebas_basicas.txt
$./pruebas_basicas.txt
```

- Muestra de ayuda

```
$ ./tp0 -h o ./tp0 --help

Usage:
  tp0 -h
  tp0 -V
  tp0 [options]

Options:
  -V, --version    Print version and quit.
  -h, --help     Print this information and quit.
  -i, --input    Location of the input file.
  -o, --output   Location of the output file.
  -a, --action   Program action: encode (default) or decode.

Examples:
  tp0 -a encode -i ~/input -o ~/output
  tp0 -a decode
```

- Muestra de version

```
$ ./tp0 -V o ./tp0 --version
Organizacion de Computadoras - TP0

Encoder/Decoder Base64 - v0.2

Group Members:
Gonzalez Perez, Ailen    Padron: 97043
Mariotti, Maria Eugenia  Padron: 96260
Ra[U+FFFD]a, Cristian Ezequiel  Padron: 95457
```

- Archivo de entrada no válido

```
$ ./tp0 -i fail.txt
```

[1] Las distintas decodificaciones/codificaciones de referencia, se obtuvieron con el codificador online www.base64encode.org y el decodificador www.base64decode.org.

```
Error de apertura de archivo de entrada. No se puede continuar.
```

3.0.2. Decodificador

La ejecución de los siguientes comandos debería dar como resultado (leído de corrido) un fragmento de **El amor en tiempos de cólera**[2]

- Ejecución de las pruebas

```
$chmod +x pruebas_decode.txt
$ ./pruebas_decode.txt
```

- Comandos que la componen

```
echo TGUgcm9nw7MgYSBEaW9zIHF1ZSBsZSBjb25jZWRpZXJhIGFsIG1lbm9zIHVuIGluc3RhbnRlIHBhcmEgcXVlIMOpb
CBubyBzZSBmdWVyYQ== | ./tp0 -a decode
echo
    c2luIHNhYmVyIGN1w6FudG8gbG8gaGFiw61hIHF1ZXJpZG8gcG9yIGVuY2ltYSBkZSBsYXMgZHVkYXMgZGUgYW1ib3Ms|
    ./tp0 -a decode
echo
    eSBzaW50acOzIHVuIGFwcmVtaW8gaXJyZXNpc3RpYmxlIGRlIGVtcGV6YXIgbGEgdmlkYSBjb24gw6lsIG90cmEgdmV6|
    ./tp0 -a decode
echo
    ZGVzZGUgZWwgcHJpbmNpcGlvIHBhcmEgZGVjaXJzZSBOb2RvIGxvIHF1ZSBzZSBsZXMgcXVlZMOzIHNpbiBkZWNpciw=|
    ./tp0 -a decode
echo
    eSB2b2x2ZXIgYSBoYWNlciBiaWVuIGN1YWxxdWllciBjb3NhIHF1ZSBodWJpZXJhbiBoZWNobyBtYWwgZW4gZWwgcGFz
YWRvLg==| ./tp0 -a decode
echo UGVybyBOdXZvIHF1ZSByZW5kaXJzZSBhbnRlIGxhIGludHJhbnNpZ2VuY2lhIGRlIGxhIG11ZXJ0ZS4=| ./tp0
    -a decode
echo U3UgZG9sb3Igc2UgZGVzY29tcHVzbyBlbiB1bmEgY8OzbGVyYSBjaWVnYSBjb250cmEgZWwgbXVuZG8s| ./tp0
    -a decode
echo eSBhdW4gY29udHJhIGVsbGEgbWlzbWEsIHkgZXNvIGxlIGluZnVuZGnDsyBlbCBkb21pbmlv| ./tp0 -a decode
echo eSBlbCB2YWxvciBwYXJhIGVuZnJlbnRhcnNlIHNvbGEgYSBzdSBzb2xlZGFkLg==| ./tp0 -a decode
```

- Texto experado

  Le rogó a Dios que le concediera al menos un instante para que él no se fuera
  sin saber cuánto lo había querido por encima de las dudas de ambos,
  y sintió un apremio irresistible de empezar la vida con él otra vez
  desde el principio para decirse todo lo que se les quedó sin decir,
  y volver a hacer bien cualquier cosa que hubieran hecho mal en el pasado.
  Pero tuvo que rendirse ante la intransigencia de la muerte.
  Su dolor se descompuso en una cólera ciega contra el mundo,
  y aun contra ella misma, y eso le infundió el dominio
  y el valor para enfrentarse sola a su soledad.

3.0.3. Codificador

La ejecución de los siguientes comandos debería dar como resultado (leído de corrido) la codificación en base 64 de un fragmento de **Rayuela**[3]

- Ejecución de las pruebas

[2] Libro de Gabriel García Marquez cuya publicación original fue en 1985
[3] Libro de Julio Cortázar cuya publicación original fue en 1963

```
$chmod +x pruebas_encode.txt
$ ./pruebas_encode.txt
```

- Líneas del texto original a codificar[4]

  Cada vez iré sintiendo menos y recordando más, pero qué es el recuerdo sino el idioma de los sentimientos, un diccionario de caras y días y perfumes que vuelven como los verbos y los adjetivos en el discurso.

- Comandos que las componen [5]

```
echo Cada vez ire sintiendo menos y recordando mas, | ./tp0
echo pero que es el recuerdo sino el idioma de los sentimientos, | ./tp0 -a encode
echo un diccionario de caras y dias y perfumes que vuelven como| ./tp0
echo los verbos y los adjetivos en el discurso.| ./tp0
```

- Resultado esperado

```
Q2FkYSB2ZXogaXLDqSBzaW50aWVuZG8gbWVub3MgeSByZWNvcmRhbmRvIG3DoXMs

cGVybyBxdcOpIGVzIGVsIHJlY3VlcmRvIHNpbm8gZWwgaWRpb21hIGRlIGxvcyBzZW50aW1pZW50b3Ms

dW4gZGljY2lvbmFyaW8gZGUgY2FyYXMgeSBkw61hcyB5IHBlcmZ1bWVzIHF1ZSB2dWVsdmVuIGNvbW=

bG9zIHZlcmJvcyB5IGxvcyBhZGpldGl2b3MgZW4gZWwgZGlzY3Vyc28u
```

3.0.4. Input-Output files

Todas se van a ejecutar cuando hagamos

```
$chmod +x pruebas_files.txt
$./pruebas_files.txt
```

- Comandos que la componen

```
./tp0 -i dText1.txt
./tp0 -i eText1.txt -a decode
./tp0 -i dText2.txt -o prueba3.txt
./tp0 -i eText2.txt -a decode -o prueba4.txt
echo Man | ./tp0 -o prueba5.txt
echo TWFu | ./tp0 -a decode -o prueba6.txt
```

- Resultado esperado

  TWFu
  Man
  Y la generación de cuatro archivos de salida, cuyo contenido será:

  - prueba3.txt: T3JnYW5pemFjacOzbiBkZSBjb21wdXRhZG9yYXM=
  - prueba4.txt: Organización de computadoras
  - prueba5.txt: TWFu
  - prueba6.txt: Man

[4]Este texto se va a leer en lìneas separadas, ya que no se pasó entero al codificador, sino por partes.
[5]Se hacen algunas pruebas con encode "por defectoz con encode "seteado. Se presentan aquí las palabras sin tilde, por cuestiones de formato, pero en el archivo se encuentran correctamente ecritas.

4. Conclusiones

- La ejecución de las pruebas, cuyos resultados se ven por consola, tiene una tardanza casi igual en ambos sistemas operativos.
- Es posible, con un correcto manejo de archivos, llevar el contenido de los mismos a cualquier formato que se requiera para otra función.

5. Anexo

5.1. Análisis de la decodificación

De aquí en adelante y hasya terminar esta sección, vamos a suponer que el ingreso fue TWFu (ejemplo del enunciado) y por lo que debemos obtener Man como resultado. Procederemos a desarrollar paso a paso las líneas correspondientes a la decodificación (explicando cada una de sus partes) y los valores obtenidos paso a paso [6][7].

```
//Primer paso
buf[0] = (tmp[0] << 2) + ((tmp[1] & 0x30) >> 4)

tmp[0] = 19 = 00010011
tmp[0] << 2 = 01001100 = 76

tmp[1] = 22 = 00010110
tmp[1] & 0x30 = 00010110 & 00110000 = 00010000 = 16
(tmp[1] & 0x30) >> 4 = 00000001 = 1

tmp[0] << 2 + (tmp[1] & 0x30) >> 4 = 76 + 1 = 77

Por lo tanto, buf[0] = 77.


//Segundo paso
buf[1] = ((tmp[1] & 0xf) << 4) + ((tmp[2] & 0x3c) >> 2)

tmp[1] = 22 = 00010110
tmp[1] & 0xf = 00010110 & 00001111 = 00000110 = 6
(tmp[1] & 0xf) << 4 = 01100000 = 96

tmp[2] = 00000101 = 5
tmp[2] & 0x3c = 00000101 & 00111100 = 00000100 = 4
(tmp[2] & 0x3c) >> 2 = 00000001 = 1

(tmp[1] & 0xf) << 4 + (tmp[2] & 0x3c) >> 2 = 97

Por lo tanto, buf[1] = 97.

//Tercer paso
buf[2] = ((tmp[2] & 0x3) << 6) + tmp[3];

tmp[2] = 00000101 = 5
tmp[2] & 0x3 = 00000101 & 00000011 = 00000001
tmp[2] & 0x3 << 6 = 01000000 = 64

tmp[3] = 46 = 00101110
```

[6] Los valores de tmp[i] se pueden verificar con la ejecución del código.
[7] Tener en cuenta: 0x30 = 48 0x3c = 60 0xf= 15 0x3 = 3

```
tmp[2] & 0x3) << 6 + tmp[3] = 01000000 + 00101110 = 01101110 = 110


Por lo tanto, buf[2] = 110 .
```

Si buscamos los valores de buf[i] con i=0,1,2 en la tabla de caracteres ASCII observamos que: 77 corresponde al caracter M
97 corresponde al caracter a
110 corresponde al caracter n

Por lo que la salida es, tal como se esperaba, **Man**.[8]

5.2. Algunas codificaciones y decodificaciones obtenidas

Se presentan a continuación una serie de codificaciones y decodificaciones [9]

- "Y sobre todo, sé fiel a ti mismo, pues de ello se sigue, como el día a la noche, que no podrás ser falso con nadie."

```
Ilkgc29icmUgdG9kbywgc8OpIGZpZWwgYSBOaSBtaXNtbywgcHVlcyBkZSBlbGxv\\
IHNlIHNpZ3VlLCBjb21vIGVsIGTDrWEgYSBsYSBub2NoZSwgcXVlIG5vIHBvZHLDo\\
XMgc2VyIGZhbHNvIGNvbiBuYWRpZS4i
```

- "Me dijeron que en el reino del revés, nada el pájaro y vuela el pez."

```
ICJNZSBkaWplcm9uIHF1ZSBlbiBlbCByZWlubyBkZWwgcmV2w6lzLCBuYWRhIGV\\
sIHDDoWphcm8geSB2dWVsYSBlbCBwZXouIg==
```

- Estrictamente hablando, base 64 es un grupo de esquemas de codificación similares."

```
IkVzdHJpY3RhbWVudGUgaGFibGFuZG8sIGJhc2UgNjQgZXMgdW4gZ3J1cG8gZGUg\\
ZXNxdWVtYXMgZGUgY29kaWZpY2FjacOzbiBzaW1pbGFyZXMuIg==
```

5.3. Código fuente y enunciado

Se adjuntan al final de este trabajo el código fuente del programa, parte del código MIPS32 generado por el compilador y el enunciado del trabajo práctico.

Además se puede acceder al código y el presente informe en el link
https://github.com/emariotti3/66.20-ORGA.

[8] Este proceso se puede repetir también para la codificación

[9] El texto en base 64 se pone en múltiples líneas para mayor claridad, pero el mismo corresponde a todos los caracteres de forma contínua.

Referencias

[1] Base64 (Wikipedia) http://en.wikipedia.org/wiki/Base64

[2] RFC 2045: Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies; secci´on 6.8, Base64 Content-TransferEncoding. http://tools.ietf.org/html/rfc2045section-6.8

[3] 15 Practical Examples of 'echo' command in Linux http://www.tecmint.com/echo-command-in-linux/

[4] GXemul, http://gavare.se/gxemul/

[5] The NetBSD project, http://www.netbsd.org/

[6] https://www.base64decode.org/

[7] http://fm4dd.com/programming/base64/base64_algorithm.htm

[8] http://ascii.cl/es/

[9] https://es.wikipedia.org/wiki/Entrada_est