Some facts about Base64 Encoding

Praseed Pai

Base64 Character Set

Index	Binary	Char	Index	Binary	Char	Index	Binary	Char	Index	Binary	Char
0	000000	A	16	010000	Q	32	100000	g	48	110000	w
1	000001	B	17	010001	R	33	100001	h	49	110001	x
2	000010	C	18	010010	S	34	100010	i	50	110010	y
3	000011	D	19	010011	T	35	100011	j	51	110011	z
4	000100	E	20	010100	U	36	100100	k	52	110100	0
5	000101	F	21	010101	V	37	100101	l	53	110101	1
6	000110	G	22	010110	W	38	100110	m	54	110110	2
7	000111	H	23	010111	X	39	100111	n	55	110111	3
8	001000	I	24	011000	Y	40	101000	o	56	111000	4
9	001001	J	25	011001	Z	41	101001	p	57	111001	5
10	001010	K	26	011010	a	42	101010	q	58	111010	6
11	001011	L	27	011011	b	43	101011	r	59	111011	7
12	001100	M	28	011100	c	44	101100	s	60	111100	8
13	001101	N	29	011101	d	45	101101	t	61	111101	9
14	001110	O	30	011110	e	46	101110	u	62	111110	+
15	001111	P	31	011111	f	47	101111	v	63	111111	/
Padding		=									

How to represent Base64 character set in Java and C/C++?

```
static byte m_base64tab[]={'A','B','C','D','E','F','G','H',
        'I','J','K','L','M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z',
        'a','b','c','d','e','f','g','h','i','j','k','l',
        'm','n','o','p','q','r','s','t','u','v','w','x','y','z','0','1','2','3','4','5'
        ,'6','7','8','9','+','/'};
```

```
#include <stdio.h>
#include <windows.h>

//////////////////////////////////////
//
// Legal Base64 characters
// Encoder looks up in the table using
// a 6 bit index.
//

BYTE Base64tab[]={

'A','B','C','D','E','F','G','H',
'I','J','K','L','M','N','O','P',
'Q','R','S','T','U','V','W','X',
'Y','Z','a','b','c','d','e','f',
'g','h','i','j','k','l', 'm','n',
'o','p','q','r','s','t','u','v',
'w','x','y','z','0','1','2','3',
'4','5','6','7','8','9','+','/'

};
```

How To Base64 Encode a File?

- Take Three Bytes at a Time From the Input File
- The 24 bits has to be chopped into 4 Six bits value
- Look it up in the Base64 table and output 4 Bytes
- Take the Next Three Bytes
- If there is some more characters left, use the padding character “=“

Encoding Three Bytes to 4 Bytes

```
//////////////////////////////////////////////////
//
// Encode three bytes (in ) into four bytes
// of data (out) .. The algorithm converts
// three bytes of data (24 bits ) into 4
// 6 bit indexes and uses the index to emit
// the character from the Base64 table into
// output (out)

void EncodeThree( BYTE *inp , BYTE *out){
        long val = inp[0];
        val = val | inp[1] << 8;
        val = val | inp[2] << 16;
        out[0] = Base64tab[val&63];
        out[1] = Base64tab[( val >> 6 )&63];
        out[2] = Base64tab[(val >> 12 )&63];
        out[3] = Base64tab[(val >> 18 ) &63];
}
```

Decoing a Base64 Triplets

```
////////////////////////////////
// Get the index of the character from
// the Base 64 table. This routine is
// used by the Decoder .. This can be
// optimized by using a Base 64 reverse
// table...
//
int ConvertAlphabetToIndex( BYTE alpha ){
        for(int i=0; i < 63; ++i )
                if ( Base64tab[i] == alpha )
                        return i;
        return -1;
}
```

```
////////////////////////////////
// Takes four bytes of Base64 encoded data and
// converts to three bytes of data ...
// Convert the Character into the index of Base
// 64 table and assemble four indexes into a
// long value and extract the lowest three bytes
//
void DecodeThree( BYTE *inp , BYTE *out ){
        long val;
        val = ConvertAlphabetToIndex(inp[0]);
        val |= ConvertAlphabetToIndex(inp[1]) << 6;
        val |= ConvertAlphabetToIndex(inp[2]) << 12;
        val |= ConvertAlphabetToIndex(inp[3]) << 18;
        out[0] = val&255;
        out[1] = (val >> 8 )& 255;
        out[2] = (val >> 16)& 255;
}
```

Entry Point!

```
////////////////////////////////////////
// A driver program....
//
int main( int argc , char **argv ){
        BYTE in[3];
        BYTE out[4];
        in[0]='B';
        in[1]='O';
        in[2]='B';
        EncodeThree(in,out);
        printf("%c%c%c%c\n",out[0],out[1],out[2],out[3]);
        DecodeThree(out,in);
        printf("%c%c%c\n",in[0],in[1],in[2]);
}
```

Whole Program available@

- https://github.com/praseedpai/ElementaryMathForProgrammingSeries/blob/master/AlgebraNArith/Base64/Base.cpp

A Java Implementation of Base64 Encode

- https://github.com/praseedpai/JavaMail2001Proto/blob/master/SMTPJAVA/MimeEncodedFile.java