

Remarque : les fonctions `main()` n'étaient pas demandées dans l'énoncé mais sont présentes dans les corrections pour illustrer.

Question 1 - 1337 5p34k (6 points)

```
#include <iostream>
using namespace std;

const int MAX_PHRASE = 128;
const int MAX_ALPHABET = 26;
typedef char Phrase[MAX_PHRASE];
typedef char Alphabet[MAX_ALPHABET];

void leetSpeak(Phrase, Alphabet);

int main()
{
    Alphabet alphabet = {'4','8','[',')','3','F','6','#','1',
                        'J','k','l','M','N','0','p','Q','2',
                        '5','7','u','V','W','X','j','5'};

    Phrase phrase = "Leet Speak";
    leetSpeak(phrase,alphabet);
    cout << phrase << endl;
    return 0;
}

void leetSpeak(Phrase s, Alphabet a)
{
    for(int i=0 ; s[i] != '\0' && i < MAX_PHRASE ; i++)
    {
        if(s[i]>='a' && s[i]<='z')
            s[i] = a[s[i] - 'a'];
        else if(s[i]>='A' && s[i]<='Z')
            s[i] = a[s[i] - 'A'];
    }
}
```

Question 2 - Tri d'un dictionnaire (8 points)

```
#include <iostream>
using namespace std;

const int N = 15;
const int NBWORDS = 1000;
typedef char Dictionnaire[NBWORDS][N];
typedef int Index[N+1];

void affiche(Dictionnaire, int);
bool estAvant(Dictionnaire, int, int, int);
void triDico(Dictionnaire , Index, int );

int main()
{
    Dictionnaire dico = { {'O'}, {'A'},
                          {'B','U'}, {'A','N'},
                          {'B','I','O'}, {'B','A','C'},
                          {'M','U','R'}, {'A','N','S'},
                          {'M','U','L','E'} };

    affiche(dico, 10);
    Index index = {0,2,4,8,9};
    triDico(dico, index, 3);
    affiche(dico, 10);
    return 0;
}

void affiche(Dictionnaire dico, int size) //n'est pas demandée à l'examen
{
    for(int i=0; i< size;i++)
        cout << dico[i] << endl;
}

bool estAvant(Dictionnaire dico, int i, int j, int taille)
{
    int k;
    for(k=0; dico[i][k] == dico[j][k] && k<taille-1; k++);
    return dico[i][k] < dico[j][k];
}

void triDico(Dictionnaire dico, Index index, int taille)
{
    for(int i=index[taille-1]; i<index[taille]-1; i++)
    {
        int min=i;
        for(int j=i+1; j<index[taille]; j++) //trouve le minimum
        {
            if(estAvant(dico, j, min, taille))
            {
                min = j;
            }
        }
        for(int k=0;k<taille;k++) //inverse (swap)
        {
            char sauvegarde = dico[i][k];
            dico[i][k] = dico[min][k];
            dico[min][k] = sauvegarde;
        }
    }
}
}
```

Question 3 - Série numérique (6 points)

```
#include <cmath>
#include <iostream>

using namespace std;
const double EPS = 1.0e-8;
const int NMAX = 100;

double serie(double alpha)
{
    double somme = 0.0;
    for ( int n = 2 ; n <= NMAX ; ++n )
    {
        somme += 1.0 / pow(n, alpha);
    }
    return somme;
}

double alpha(double cible)
{
    double bi = 1.0;
    double bs = 100.0;
    double alpha,somme;

    do {
        alpha = (bs + bi) / 2.0;
        somme = serie(alpha);
        if (somme<cible)
            bs = alpha;
        else
            bi = alpha;
    } while (abs(somme - cible) > EPS);

    return alpha;
}

int main()
{
    cout << alpha(0.5) << " / " << serie(alpha(0.5)) <<endl;
    return 0;
}
```