4.1. Introduction

Dans ce chapitre on essaye d'implémenter l'ensemble des techniques permettant de **chiffrer** des textes arabes, c'est-à-d permettant de les rendre inintelligibles sans une action spécifique.

4.2. Objectif

Dans notre application nous avons tenté à voir les différents algorithmes de cryptographie appliquée aux textes arabes savoir une meilleure qualité de protection de messages.
Parmi ces algorithmes on a choisi d'implémenter l'algorithme à clé secrète DES, et l'algorithme à clé publique le RSA.

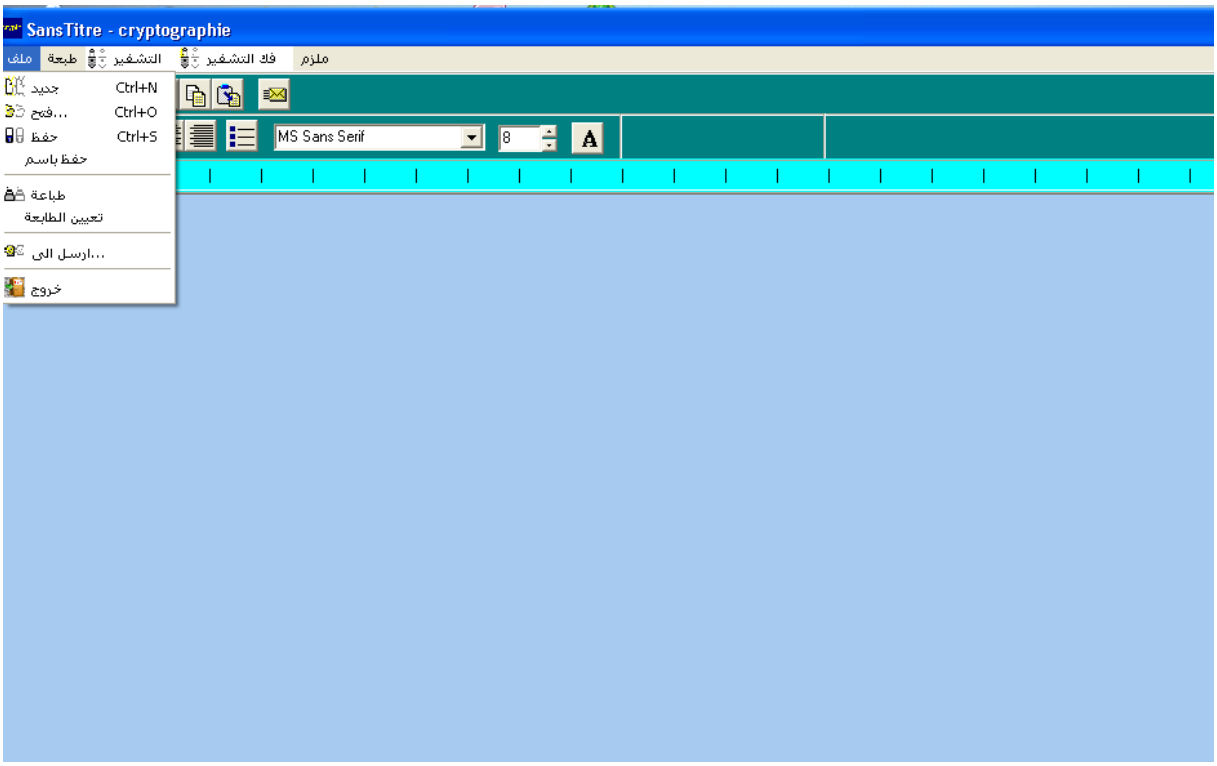
4.3. Logiciel utilisé

C++ builder 6

Le langage choisi pour réalisation de notre application est le **BORLAND C++ BUILDER6.** Ce choix repose sur le fait que Borland possède tout la puissance du langage C++ oriente objet comme il offre la possibilité de développer rapidement des applications sous Windows grâce a ses différentes bibliothèques. Il permet la création instantanée des interfaces utilisateurs car il offre une gestion de l'interface.

4.3.1. Description de l'interface et composantes

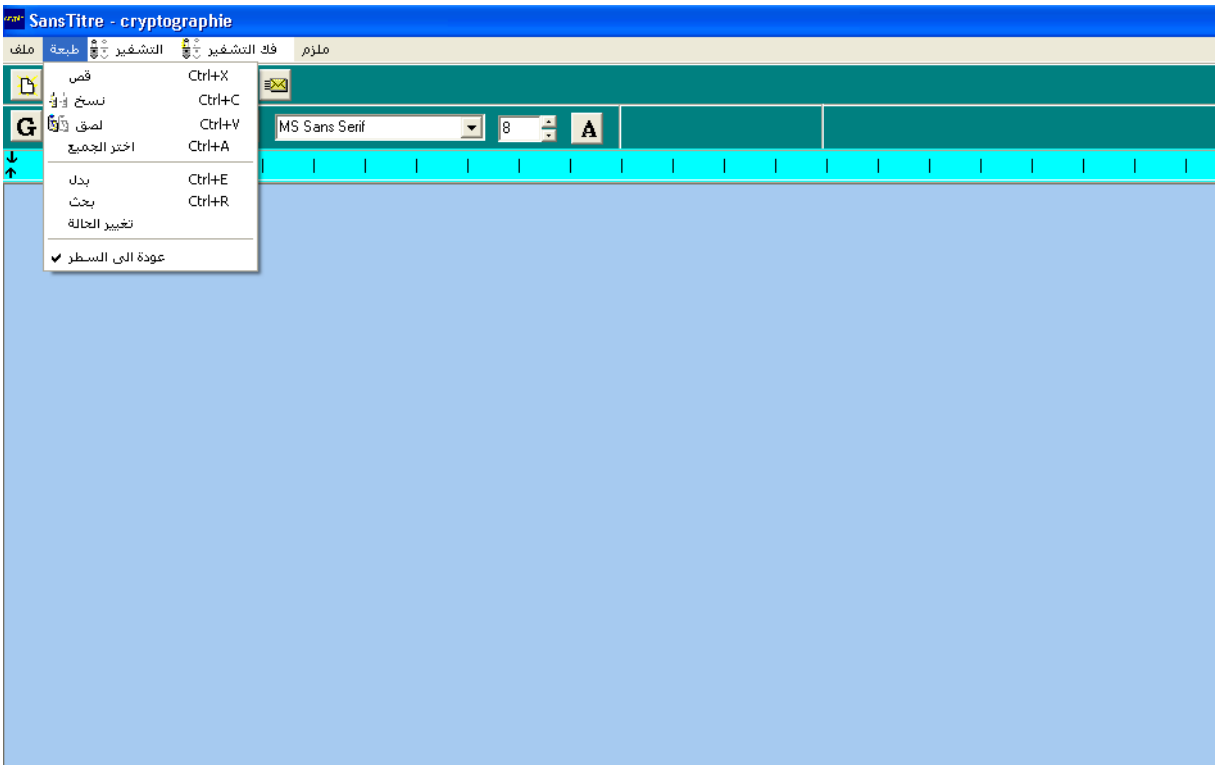
4.4. Le contenu du menu « ملف»



- **جديد**: une nouvelle page.
- **فتح**: ouvrir un texte arabe .
- **حفظ**: enregistrer le texte.
- **حفظ باسم**: enregistrer sous.

- **طباعة**: imprimer la page.
- **تعيين الطابعة**: choisir une imprimante.
- **أرسل إلى**: envoyer vers.
- **خروج**: sort du programme.

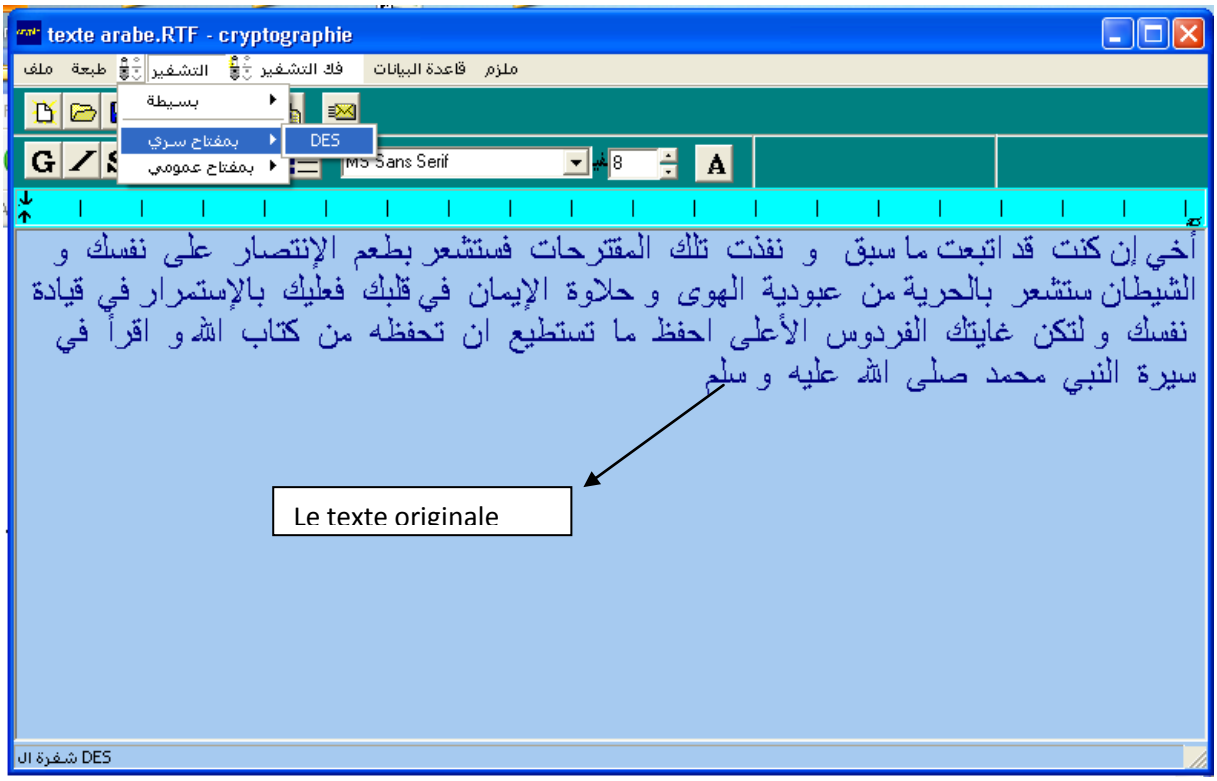
4.5. Le contenu du menu « طبعة »



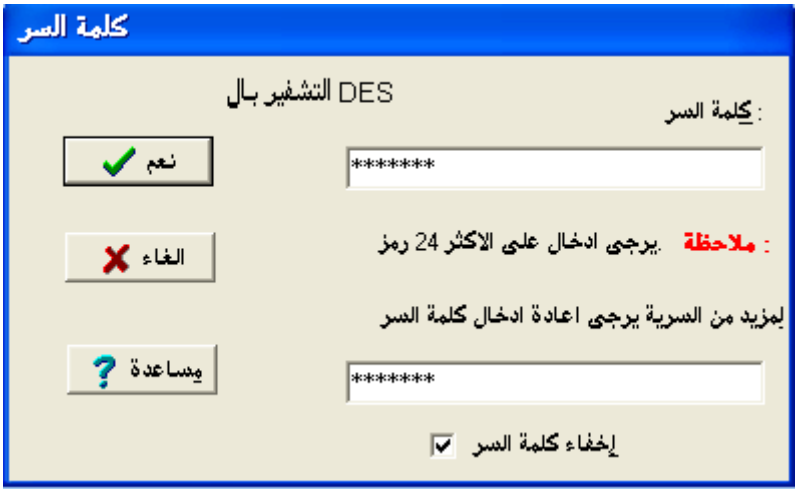
- قص :couper.
- نسخ:copier.
- لصق:coller.
- اختر الجميع:sélectionner tout.
- بحث: chercher.
- بدل:changer.
- تغيير الحالة:modifier.
- عودة إلى السطر:retour a la ligne

4.6. Exemple de quelque opérateur

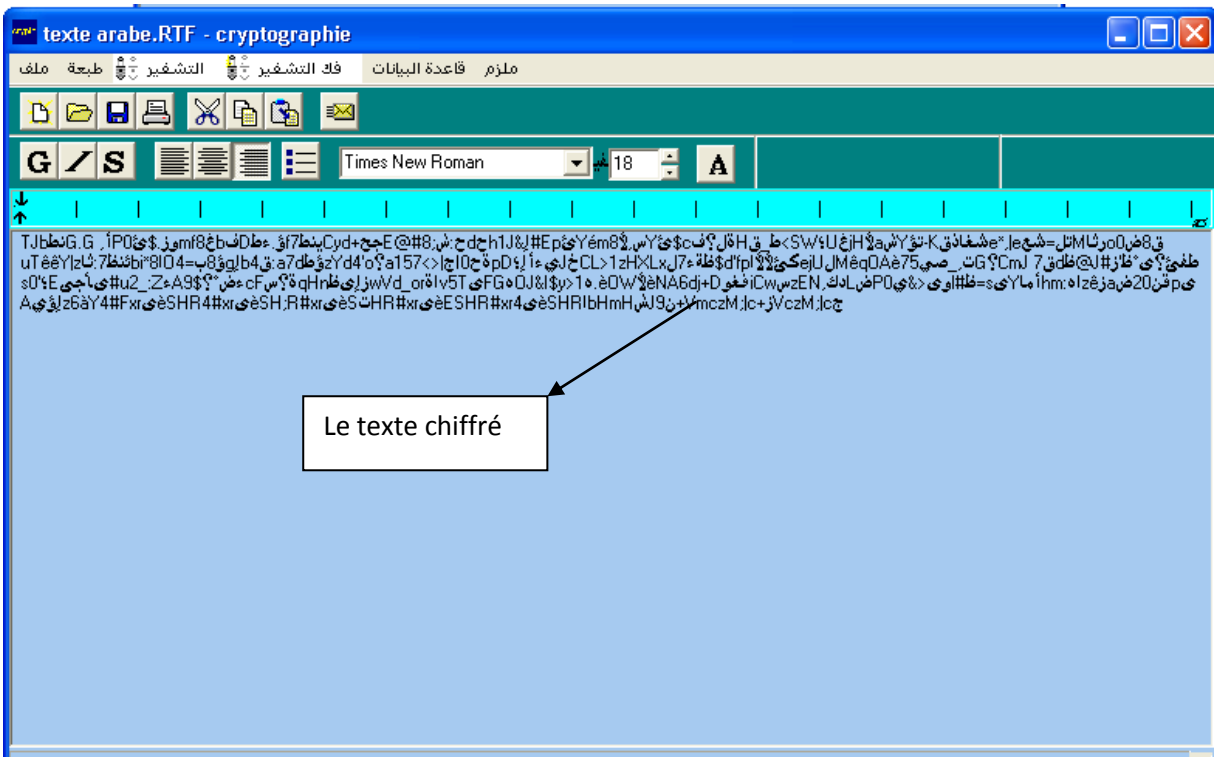
4.6.1. Exemples de chiffrement à clé secrète (DES)



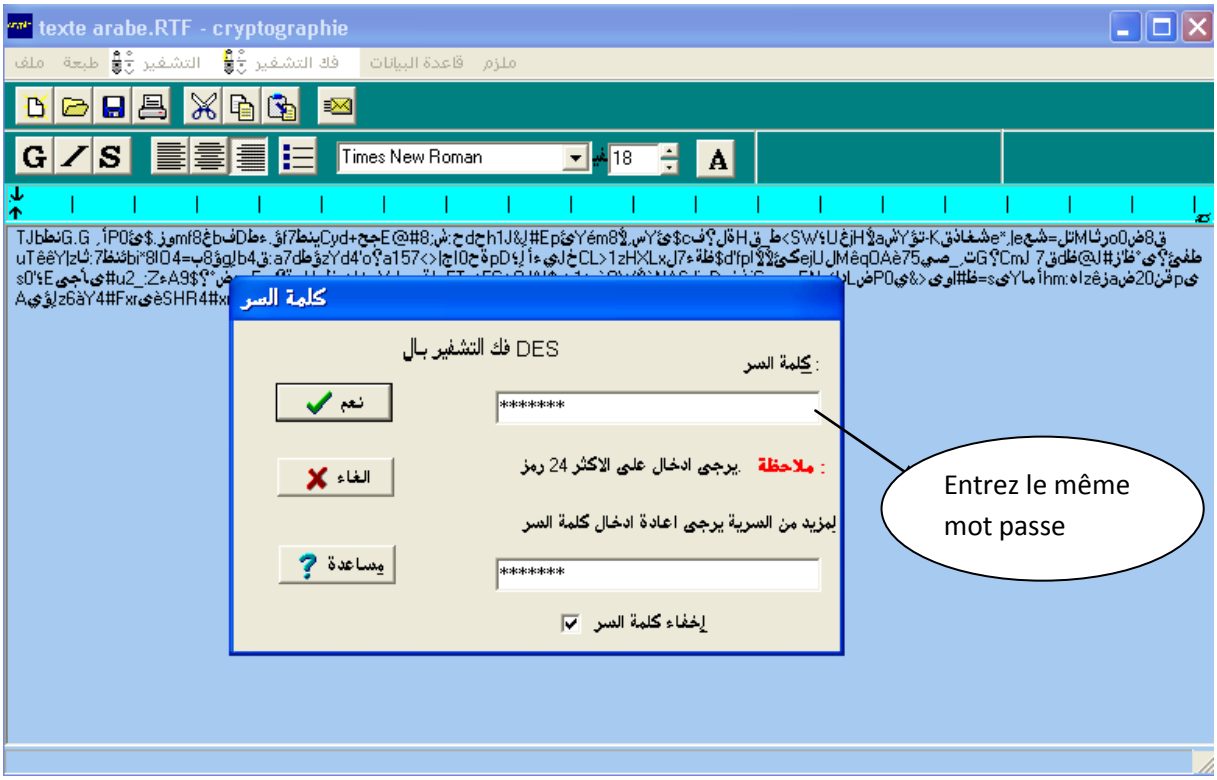
Introduction de la clé:

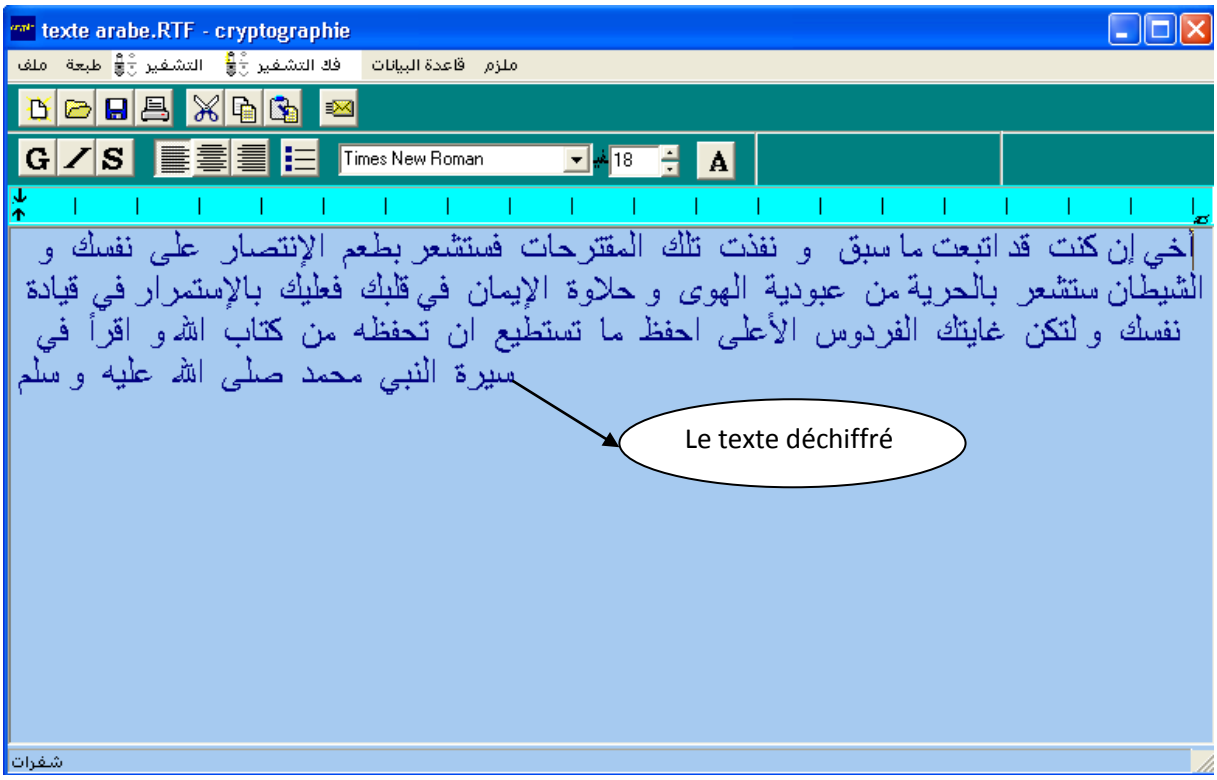


Le texte chiffré :



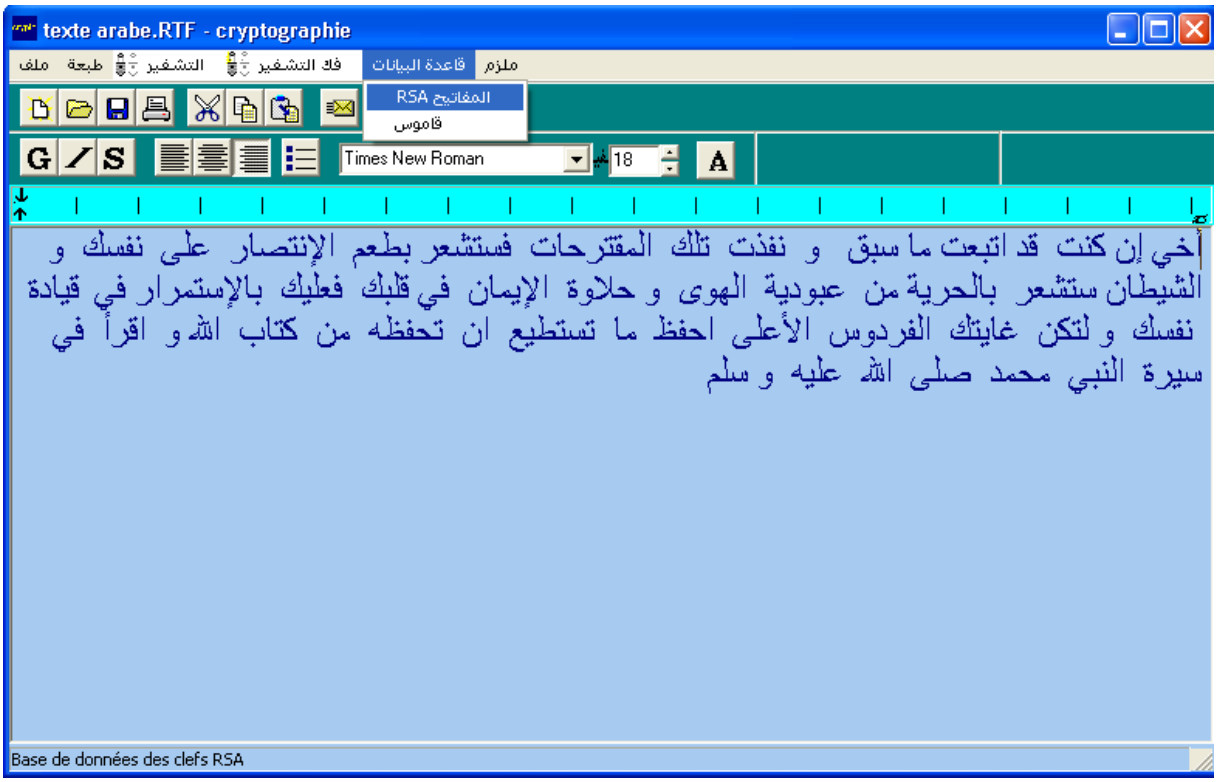
Pour déchiffrer

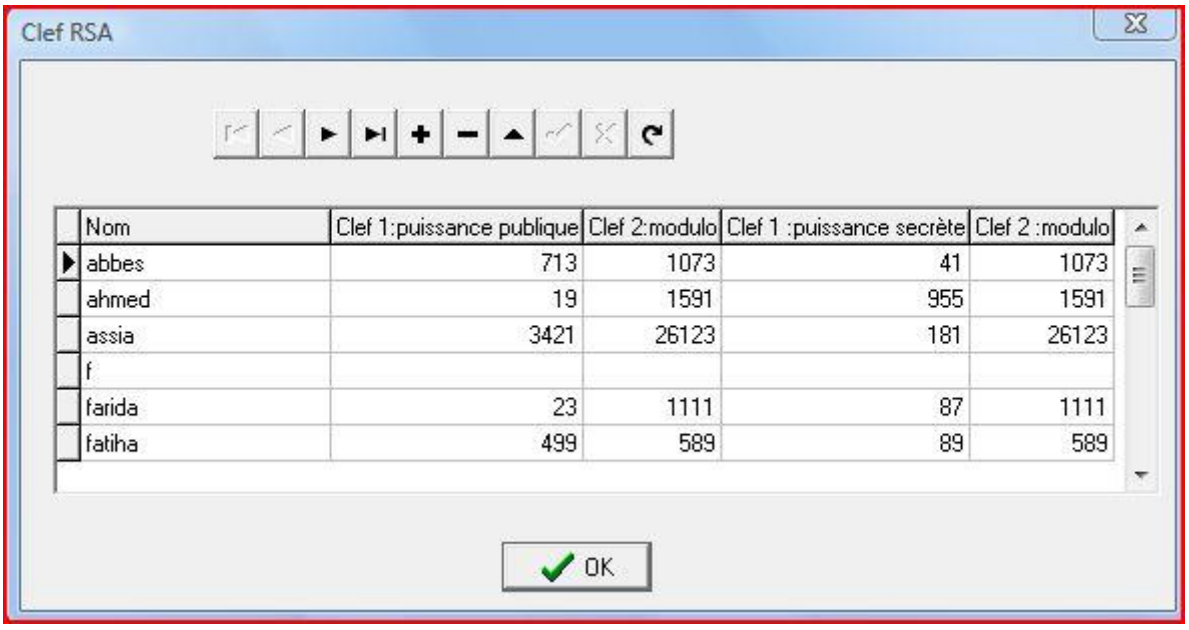




4.6.2. Exemple de chiffrement à clé publique (RSA)

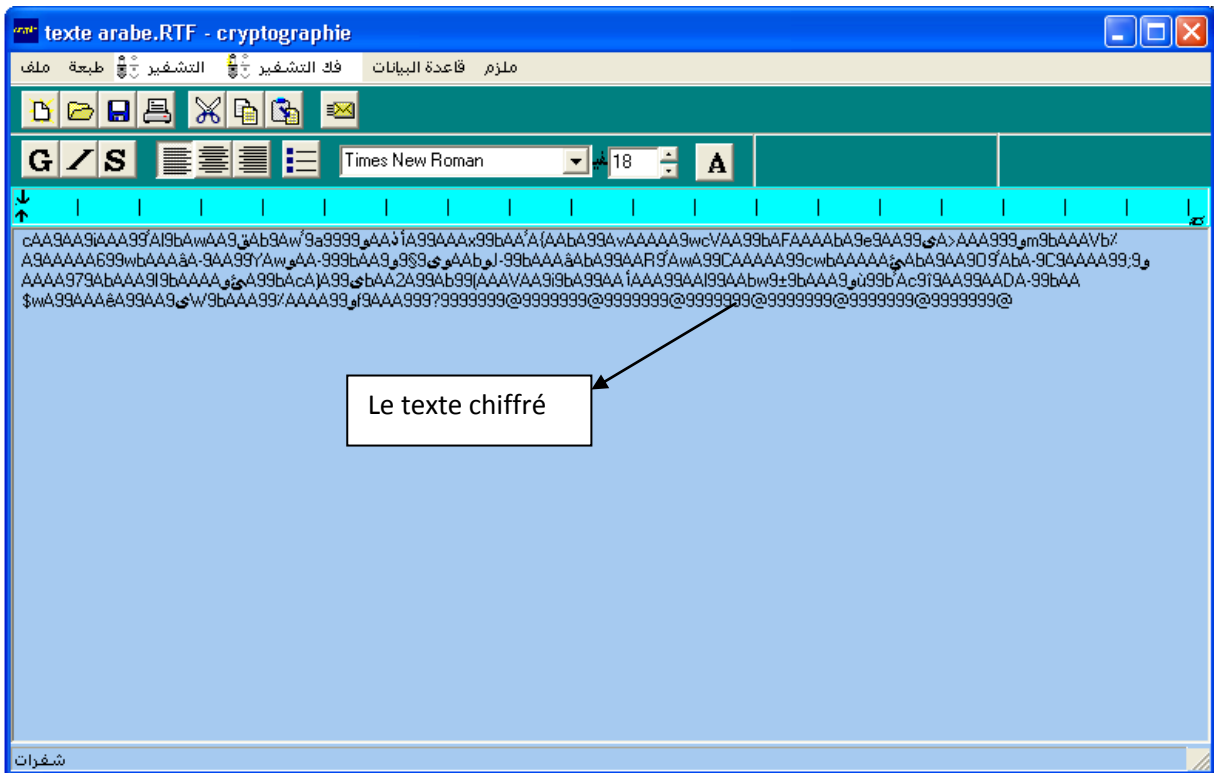
Les clefs RSA :



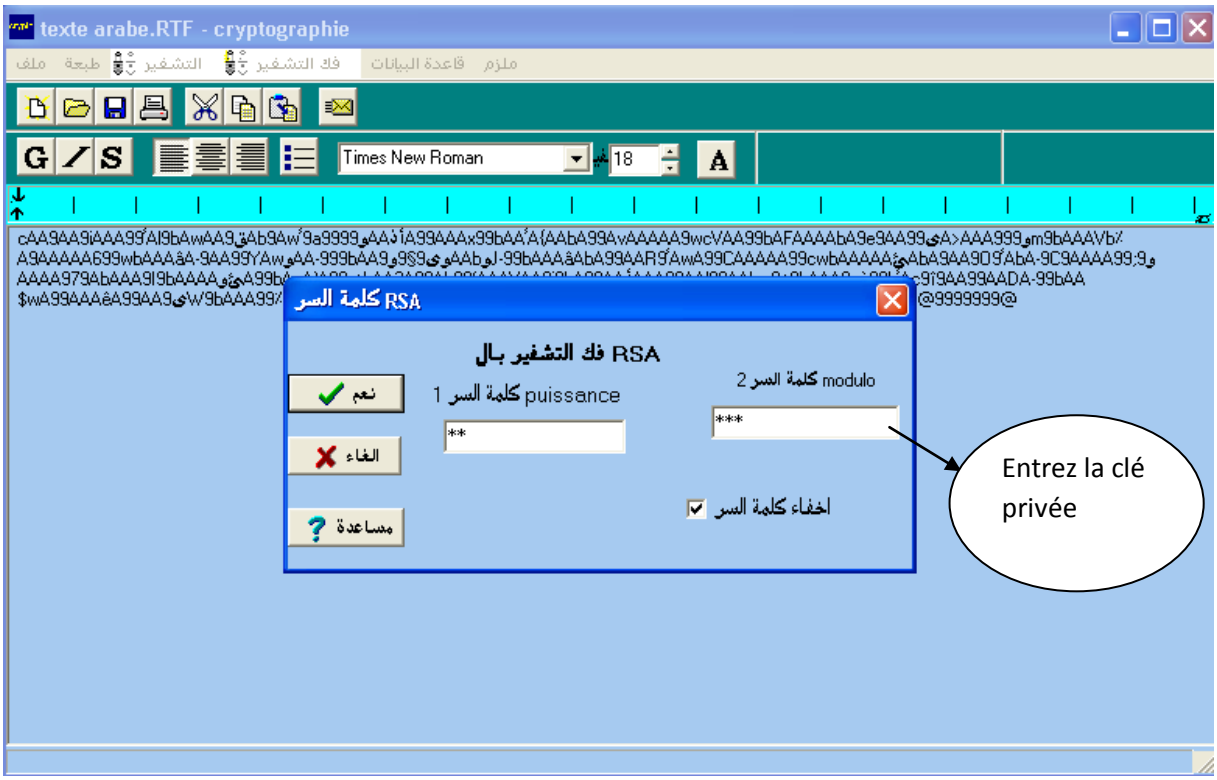


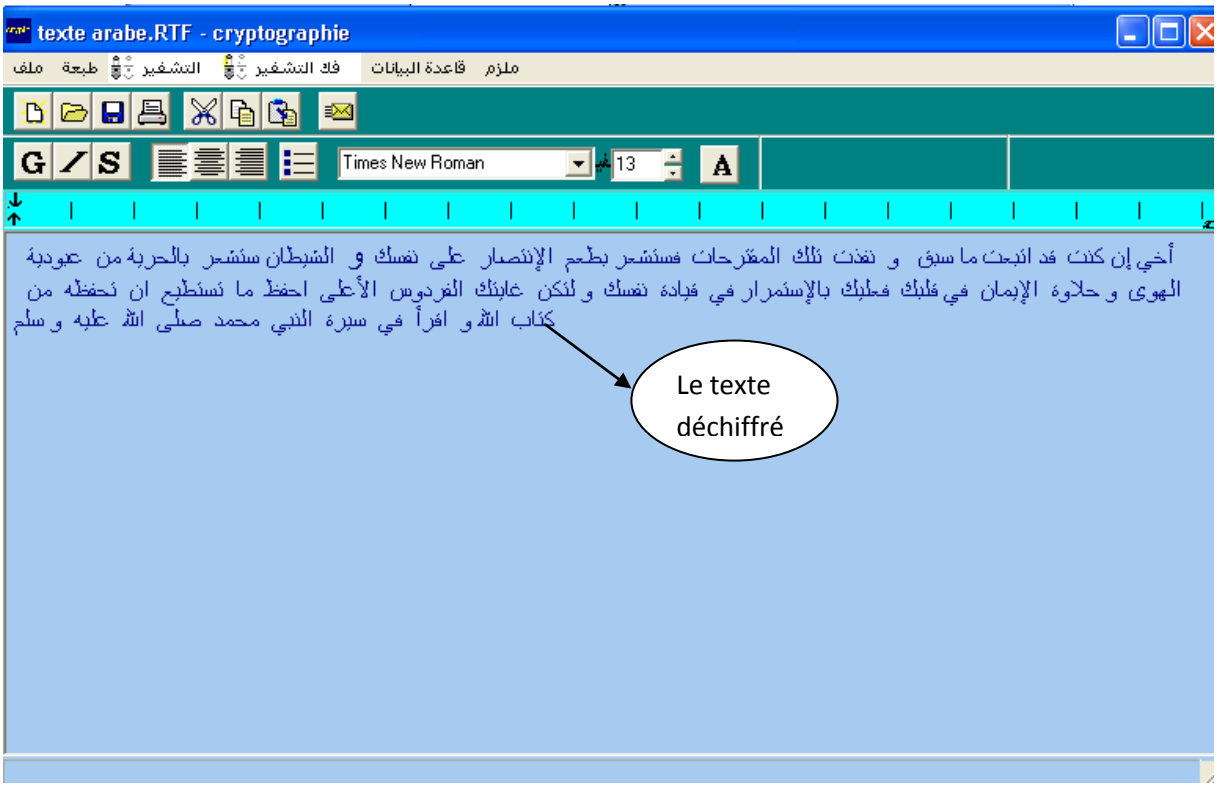
La fenêtre de clé publique entrer :

Le texte chiffré :



Pour déchiffrer


texte arabe.RTF - cryptographie
ملف طبعة التشفير فك التشفير قاعدة البيانات ملزم
Times New Roman
18
كلمة السر RSA
فك التشفير بال RSA
نعم
الغاء
مساعدة
puissance كلمة السر 1
**
modulo كلمة السر 2
***
اخفاء كلمة السر
Entrez la clé privée


texte arabe.RTF - cryptographie
ملف طبعة التشفير فك التشفير قاعدة البيانات ملزم
Times New Roman
13
أخي إن كنت قد اتبعت ما سبق و نفذت تلك المقترحات فستشعر بطعم الإنتصار على نفسك و الشيطان ستشعر بالحرية من عبودية الهوى و حلاوة الإيمان في قلبك فعليك بالإستمرار في قيادة نفسك و لتكن غايتك الفردوس الأعلى احفظ ما تستطيع ان تحفظه من كتاب الله و اقرأ في سيرة النبي محمد صلى الله عليه و سلم
Le texte déchiffré

4.7. Lettre Arabe

Utilisés pour l'écriture arabe.

Les caractères U+0600 à U+0603 et U+06DD sont des signes de contrôle de format.

Les caractères U+0610 à U+0615, U+064B à U+065E, U+0670, U+0, U+06D6 à U+06DC, U+06DF à U+06E4, U+06E7, U+06E8 et U+06EA à U+06ED sont des signes diacritiques se combinant avec le caractère qu'ils suivent ; ils sont combinés ici avec la lettre arabe *sīn* « س » (U+0633) à des fins de lisibilité.

Note : certaines polices de caractères arabes indiquent supporter tout ce sous-ensemble de caractères, mais n'affichent aucun glyphe pour certains d'entre eux.

Table des caractères

voir PDF : fr en	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
060	ـــ	ـــ	ـــ	ـــ								ف	،	ـ	ـــ	ع
061	سـً	سـَ	سـُ	سـِ	سـّ	سـْ						؛			؞	؟
062		ء	آ	أ	ؤ	إ	ئ	ا	ب	ة	ت	ث	ج	ح	خ	د
063	ذ	ر	ز	س	ش	ص	ض	ط	ظ	ع	غ					
064	ـ	ف	ق	ك	ل	م	ن	ه	و	ى	ي	سـً	سـٌ	سـٍ	سـَ	سـُ
065	سـِ	سـّ	سـْ	سـٓ	سـٔ	سـٕ	سـٖ	سـٗ	سـ٘	سـٙ	سـٚ	سـٛ	سـٜ	سـٝ	سـٞ	
066	٠	١	٢	٣	٤	٥	٦	٧	٨	٩	٪	،	،	*	ـ	ـ
067	سـٰ	سـٱ	أ	إ	ٴ	ٵ	ؤ	ؤ	ئ	ٹ	ٺ	ٻ	ټ	ټ	پ	ټ
068	ڀ	ځ	څ	ڃ	ڃ	څ	چ	چ	ڈ	ډ	ډ	ڈ	ډ	ډ	ډ	ډ
069	ڈ	ڑ	ڑ	ڑ	ڑ	ڑ	ڑ	ڑ	ژ	ڑ	ښ	ڜ	ڜ	ڝ	ڞ	ڟ
06A	ڠ	ف	ڤ	ڤ	ڤ	ڤ	ڤ	ف	ق	ک	ک	ک	ك	ڪ	ګ	گ
06B	گ	گ	گ	گ	گ	ڵ	ڵ	ڵ	ڵ	ڼ	ں	ڻ	ڼ	ڼ	ھ	ڃ
06C	ۀ	ه	ۀ	ة	ۄ	ۄ	ۆ	ۇ	ۈ	ۋ	ۅ	ۋ	ى	ۍ	ێ	ۆ
06D	ي	ي	ے	ۓ	۔	ه	ۘ	ۗ	ۙ	ۚ	ۛ	ۜ	۝	۞	۟	ۘ
06E	۠	ۡ	ۢ	ۣ	ۤ	ۥ	ۦ	ۧ	ۨ	۩	۪	۫	۬	ۭ	ۮ	ۯ
06F	٠	١	٢	٣	۴	۵	۶	۷	۸	۹	ۺ	ۻ	ۼ	۽	۾	ۿ

4.7.1. Table des caractères ASCII

La première colonne de cette table renferme le code MARC à 8 bits (en hex) pour le caractère tel que provenant du jeu graphique G0; la seconde colonne de cette table renferme le code MARC à 8 bits (en hex) pour le caractère tel que provenant du jeu graphique G1; la troisième colonne contient le code UCS/Unicode à 16-bits (en hex), et la quatrième colonne contient le code UTF-8 (en hex) pour les caractères UCS; la cinquième colonne contient une image d'une représentation du caractère; la sixième colonne indique les noms des caractères : nom MARC / nom UCS. Si le nom MARC est le même que le nom UCS, ou s'il est semblable à ce dernier, seul le nom UCS apparaît.

MARC-8 jeu G0	MARC-8 jeu G1	UCS	UTF-8	Caractères	Nom MARC / Nom UCS
21	A1	0021	21	!	Point d'exclamation
22	A2	0022	22	"	Guillemet anglais
23	A3	0023	23	#	Symbole de numéro
24	A4	0024	24	$	Symbole de dollar
25	A5	066A	D9AA	٪	Signe pour cent / Signe pour cent arabe
26	A6	0026	26	&	Perluète
27	A7	0027	27	'	Apostrophe
28	A8	0028	28	(	Parenthèse ouvrante / Parenthèse gauche
29	A9	0029	29	)	Parenthèse fermante / Parenthèse droite
2A	AA	066D	D9AD	٭	Astérisque / Étoile à cinq pointes arabe
2B	AB	002B	2B	+	Signe plus
2C	AC	060C	D88C	،	Virgule arabe
2D	AD	002D	2D	-	Trait-d'union - signe moins
2E	AE	002E	2E	.	Point, point décimal / Point
2F	AF	002F	2F	/	Barre oblique / Cotice
30	B0	0660	D9A0	٠	Chiffre zéro arabe-indo-aryen / Chiffre arabe-hindi zéro
31	B1	0661	D9A1	١	Chiffre un arabe-indo-aryen / Chiffre arabe-hindi un
32	B2	0662	D9A2	٢	Chiffre deux arabe-indo-aryen / Chiffre arabe-hindi deux
33	B3	0663	D9A3	٣	Chiffre trois arabe-indo-aryen / Chiffre arabe-hindi trois
34	B4	0664	D9A4	٤	Chiffre quatre arabe-indo-aryen / Chiffre arabe-hindi quatre
35	B5	0665	D9A5	٥	Chiffre cinq arabe-indo-aryen / Chiffre arabe-hindi cinq

36	B6	0666	D9A6	٦	Chiffre six arabe-indo-aryen / Chiffre arabe-hindi six
37	B7	0667	D9A7	٧	Chiffre sept arabe-indo-aryen / Chiffre arabe-hindi sept
38	B8	0668	D9A8	٨	Chiffre huit arabe-indo-aryen / Chiffre arabe-hindi huit
39	B9	0669	D9A9	٩	Chiffre neuf arabe-indo-aryen / Chiffre arabe-hindi neuf
3A	BA	003A	3A	:	Deux points
3B	BB	061B	D89B	؛	Point virgule arabe
3C	BC	003C	3C	<	Signe inférieur à
3D	BD	003D	3D	=	Signe égal à
3E	BE	003E	3E	>	Signe supérieur à
3F	BF	061F	D89F	؟	Point d'interrogation arabe
41	C1	0621	D8A1	ء	Hamzah / Lettre arabe hamza
42	C2	0622	D8A2	آ	Lettre arabe alef avec madda au-dessus / Lettre arabe alif madda en chef
43	C3	0623	D8A3	أ	Lettre arabe alef avec hamza au-dessus / Lettre arabe alif hamza en chef
44	C4	0624	D8A4	ؤ	Lettre arabe waw avec hamza au-dessus / Lettre arabe waw hamza en chef
45	C5	0625	D8A5	إ	Lettre arabe alef avec hamza en dessous / Lettre arabe alif hamza souscrit
46	C6	0626	D8A6	ئ	Lettre arabe yeh avec hamza au-dessus Lettre arabe ya' hamza en chef
47	C7	0627	D8A7l	ا	Lettre arabe alef / Lettre arabe alif
48	C8	0628	D8A8	ب	Lettre arabe beh / Lettre arabe ba'
49	C9	0629	D8A9	ة	Lettre arabe the marbuta / Lettre arabe té' marbouta
4A	CA	062A	D8AA	ت	Lettre arabe the / Lettre arabe té'
4B	CB	062B	D8AB	ث	Lettre arabe theh / Lettre arabe thé'

4C	CC	062C	D8AC	ج	Lettre arabe jeem / Lettre arabe djîm
4D	CD	062D	D8AD	ح	Lettre arabe hah / Lettre arabe ha'
4E	CE	062E	D8AE	خ	Lettre arabe khah / Lettre arabe kha'
4F	CF	062F	D8AF	د	Lettre arabe dal / Lettre arabe dal
50	D0	0630	D8B0	ذ	Lettre arabe thal / Lettre arabe dhal
51	D1	0631	D8B1	ر	Lettre arabe reh / Lettre arabe ra'
52	D2	0632	D8B2	ز	Lettre arabe zain / Lettre arabe zaïn
53	D3	0633	D8B3	س	Lettre arabe seen / Lettre arabe sîn
54	D4	0634	D8B4	ش	Lettre arabe sheen / Lettre arabe chîn
55	D5	0635	D8B5	ص	Lettre arabe sad / Lettre arabe çad
56	D6	0636	D8B6	ض	Lettre arabe dad
57	D7	0637	D8B7	ط	Lettre arabe tah / Lettre arabe ta'
58	D8	0638	D8B8	ظ	Lettre arabe zah / Lettre arabe zza'
59	D9	0639	D8B9	ع	Lettre arabe ain / Lettre arabe 'aïn
5A	DA	063A	D8BA	غ	Lettre arabe ghain / Lettre arabe ghaïn
5B	DB	005B	5B	[	Crochet ouvrant / Crochet de gauche
5D	DD	005D	5D	]	Crochet fermant / Crochet de droite

60	E0	0640	D980	ـ	Tatweel arabe / Tatouil arabe
61	E1	0641	D981	ف	Lettre arabe feh / Lettre arabe fa'
62	E2	0642	D982	ق	Lettre arabe qaf / Lettre arabe qaf
63	E3	0643	D983	ك	Lettre arabe kaf / Lettre arabe kaf
64	E4	0644	D984	ل	Lettre arabe lam / Lettre arabe lam
65	E5	0645	D985	م	Lettre arabe meem / Lettre arabe mîm
66	E6	0646	D986	ن	Lettre arabe noon / Lettre arabe noûn
67	E7	0647	D987	ه	Lettre arabe heh / Lettre arabe hé'
68	E8	0648	D988	و	Lettre arabe waw
69	E9	0649	D989	ى	Lettre arabe alef maksura / Lettre arabe alif maksoura
6A	EA	064A	D98A	ي	Lettre arabe yeh / Lettre arabe ya'
6B	EB	064B	D98B	ً	Fathatan arabe
6C	EC	064C	D98C	ٌ	Dammatan arabe
6D	ED	064D	D98D	ٍ	Kasratan arabe

6E	EE	064E	D98E	َ	Fatha arabe
6F	EF	064F	D98F	ُ	Damma arabe
70	F0	0650	D990	ِ	Kasra arabe
71	F1	0651	D991	ّ	Shadda arabe / Chadda arabe
72	F2	0652	D992	ْ	Sukun arabe / Soukoun arabe
73	F3	0671	D9B1	ٱ	Lettre arabe alef wasla / Lettre arabe alif wasla
74	F4	0670	D9B0	ٰ	Lettre arabe majuscule alef / Lettre arabe alif en chef
78	F8	066C	D9AC	٬	Séparateur milliers arabe
79	F9	201D	E2809D	"	Guillemet double droit
7A	FA	201C	E2809C	"	Guillement double gauche

Table 4.1 : table de codage ASCII

Voila la table de codage des lettres qu'on a utilisée dans notre implémentation

```
struct cde { char lettre; unchar entier;};
struct cde table1[]={
{'٧،'ئ'}   ،{١،'ط'}   ،{٥،'ص'}   ،{٤،'ض'}   ،{٣،'ذ'}   ،{٢،'ى'}   ،{١،'ق'},
{'١٤،'ع'}  ،{١٣،'غ'}  ،{١٢،'ف'}  ،{١١،'ق'}  ،{١٠،'ث'}   ،{٩،'أ'}  ،{٨،'ظ'},
{'٢١،'س'}  ،{٢٠،'ش'}  ،{١٩،'د'}  ،{١٨،'ج'}  ،{١٧،'ح'}  ،{١٦،'ل'}  ،{١٥،'ه'},
{'٢٨،'ك'}  ،{٢٧،'م'}  ،{٢٦،'ن'}  ،{٢٥،'ت'}  ،{٢٤،'ا'}  ،{٢٣،'خ'}  ،{٢٢،'ي'},
{'٣٥،'ة'} ،{٣٤،'ـى'} ،{٣٣،'لا'}  ،{٣٢،'ر'}  ،{٣١،'ؤ'}  ،{٣٠،'ء'}  ،{٢٩،'ئ'},
{'٣٩،'¦'}  ،{٣٨،'ظ'}  ،{٣٧،'ز'}  ،{٣٦،'و'},  {'a',40},  {'b',41},  {'c',42},
{'d',43},  {'e',44},  {'f',45},  {'g',46},  {'h',47},  {'i',48},  {'j',49},
{'k',50},  {';',51},  {'à',52},  {'l',53},  {'m',54},  {'n',55},  {'o',56},
{'p',57},  {'q',58},  {'é',59},  {'è',60},  {'r',61},  {'s',62},  {'t',63},
{'u',64},  {'v',65},  {'w',66},  {'\'',67}, {' ',68},  {'x',69},  {'y',70},
{'z',71},  {'A',72},  {'B',73},  {'C',74},  {'D',75},  {',',0},   {'E',76},
{'K',77},  {'L',78},  {'M',79},  {'N',80},  {'٨٧،'.'}  ،{٨١،'ب'},  {'O',111},
{'ê',116}, {'P',83},  {'Q',84},  {'R',85},  {'S',86},  {'T',87},  {'U',88},
{'V',89},  {'W',90},  {'X',91},  {'Y',92},  {'Z',93},  {'F',94},  {'G',95},
{'H',96},  {'I',97},  {'J',98},  {'1',99},  {'2',100}, {'3',101}, {'4',102},
{'5',103}, {'6',104}, {'7',105}, {'8',106}, {'9',107}, {'0',108}, {'_',109},
{'@',110}, {'^',161}, {'$',112}, {'*',113}, {'+',114}, {'-',115}, {'|',117},
{':',118}, {'١٢٤،'#'} ،{١٢٣،'؟'} ،{١٢٢،'<'} ،{١٢١،'='} ،{١٢٠،'>'} ،{١١٩،'؛'},
{'°',125}, {'&',126}, {'ª',127}, {'"',128}, {'%',129}, {'!',130}, {'«',131},
{'»',132}, {'±',133}, {'/',134}, {'{',135}, {'!',136}, {')',137} ,{'،',138},
{'[',139}, {']',140}, {'(',141}, {'¶',142}, {'ç',143}, {'¨',144}, {'£',145},
{'¤',146}, {'µ',147}, {'§',148}, {'ù',149}, {',',150}, {'û',151}, {'î',152},
{'ô',153}, {'½',154}, {'©',155}, {'١٥٦،'}, {'â',157}, {'ï',158}, {'ˡ,159},
{'١٦٠،'Ï'}};
```

Table 4.2 table de codage des lettres

4.8. Code source de différentes implémentations

4.8. 1. Code source de l'algorithme DES

Chiffrement par l'algorithme DES

```
void __fastcall TMainForm::DES1Click( TObject *Sender)
{
//chiffrement DES
Form1->Label5->Caption="";
Form1->Label5->Caption="التشفير بال DES";
Form1->ShowModal();
// texte vide
int S = RichEdit1->GetTextLen();
if((S==0)&&(Accepter)){
         Accepter=false;
         ShowMessage("لا يمكن تشفير نص فارغ");
         }
//***
if(Accepter){
int star=clock();
int i,j,k;
int Size = RichEdit1->GetTextLen();
char *Buffer = new char[Size+1];
RichEdit1->GetTextBuf(Buffer,Size+1);
RichEdit1->Text="";
unchar t1[8], t2[8];

int n56=Size/56;
if (Size % 56 != 0) n56 = (n56+1)*56;
        else n56 = Size;
unchar *tab1 = new unchar[n56+1];
unchar *tab2 = new unchar[n56+1];

ProgressBar1->Visible=true;
ProgressBar1->Position=0;
ProgressBar1->Min=0;
ProgressBar1->Max=4;

for (i=0;i<n56;i++)  tab1[i]= 0;
```

```
for (i=0;i<Size;i++){
            j=0;
            while (Buffer[i]!=table1[j].lettre && j++<160);
            tab1[i]=table1[j].entier;
};
delete Buffer;

// Chiffrement
byte Clef[8];
for (i=0;i<8;i++) Clef[i]=0;

if (taille_pass > 8) taille_pass=8;
for (i=0;i<taille_pass;i++){
            j=0;
            while (pass[i]!=table1[j].lettre && j++<160);
            Clef[i]=table1[j].entier;
};
ProgressBar1->Position=1;
/* générateur aléatoire */

/* L'extension de la clef */
deskey(Clef,0);
chiffrer_des(n56,tab1,tab2);
ProgressBar1->Position=2;
// fin du chiffrement

// pour eviter un eventuel pb d'affichage.
int som;
som = n56/7;
som++;
unchar *TabEtend = new unchar[n56+som+1];
```

```
int enc=0;
i=0;k=0;
while(i<n56){
    for(j=0;j<7;j++,i++)  t1[j]= tab2[i];//le tableau contenat les ent chiffres
    de7vers8(t1,t2);enc++;
    for(j=0;j<8;j++,k++) TabEtend[k] = t2[j];
};
delete tab1;
delete tab2;
ProgressBar1->Position=3;
char *les_cars = new char[n56+enc+1];
for (i=0;i<n56+enc;i++){
       j=0;
       while ( ( TabEtend[i]!=table1[j].entier) && j++<160);
       les_cars[i]= table1[j].lettre;
};
les_cars[n56+enc]='\0';
ProgressBar1->Position=4;
RichEdit1->Text = les_cars;
delete les_cars;
delete TabEtend;
ProgressBar1->Visible=false;
int end=clock();
Label2->Caption="";
Label2->Caption=((end-star)*0.001);
};
}
```

déchiffrement DES

```
void __fastcall TMainForm::DES2Click(TObject *Sender)
{
/* Déchiffrement DES */
Form1->Label5->Caption="";
Form1->Label5->Caption="فك التشفير بال DES";
Form1->ShowModal();
/* texte vide*/
int S = RichEdit1->GetTextLen();
if((S==0)&&(Accepter)){
          Accepter=false;
          ShowMessage("لا يمكن فك تشفير نص فارغ");
          }

if(Accepter){
int star=clock();
int i,j,k;
int Size2 = RichEdit1->GetTextLen();

char *Buffer2 = new char[Size2+1];
RichEdit1->GetTextBuf(Buffer2,Size2+1);
RichEdit1->Text="";
unchar t3[8],t4[8];

/*
int n64=Size2/64;
if (Size2 % 64 != 0) n64 = (n64+1)*64;
        else n64 = Size2;
*/

int som2;
som2 = Size2/8;
```

```
unchar *tab2 = new unchar[Size2+1];
// barra de progression
ProgressBar1->Visible=true;
ProgressBar1->Position=0;
ProgressBar1->Min=0;
ProgressBar1->Max=4;

for (i=0;i<Size2;i++)  tab2[i]= 0;

for (i=0;i<Size2;i++){
           j=0;
           while (Buffer2[i]!=table1[j].lettre && j++<160);
           tab2[i]=table1[j].entier;
};
delete Buffer2;
//maintenant intervient ma fonction avancee
unchar *TabRes = new unchar[Size2+1];
i=0;k=0;
while(i<Size2)
{
    for(j=0;j<8;j++,i++)  t3[j]= tab2[i];
    de8vers7(t3,t4);
    for(j=0;j<7;j++,k++) TabRes[k] = t4[j];
};
ProgressBar1->Position=1;
// Déchiffrement

unchar *tab3 = new unchar[k+1];

byte Clef2[8];
for (i=0;i<8;i++) Clef2[i]=0;
if (taille pass > 8) taille pass=8;
```

```
for (i=0;i<taille_pass;i++)
{
            j=0;
            while (pass[i]!=table1[j].lettre && j++<160);
            Clef2[i]=table1[j].entier;
};
ProgressBar1->Position=2;


deskey(Clef2,1);
chiffrer_des(k,TabRes,tab3);
ProgressBar1->Position=3;
for( i=0; i<k; i++)
        tab3[i] =(unchar)(tab3[i]&0x7f);


char *les_cars2 = new char[Size2-som2+1];
for (i=0;i<Size2-som2;i++){
       j=0;
       while((tab3[i]!=table1[j].entier) && j++<160);
       les_cars2[i]= table1[j].lettre;
};
les_cars2[Size2-som2]='\0';
ProgressBar1->Position=4;
RichEdit1->Text = les_cars2;
delete les_cars2;
delete tab2;
delete tab3;
ProgressBar1->Visible=false;
int end=clock();
Label2->Caption="";
Label2->Caption=((end-star)*0.001);
};}
```

4.8.2. Code source de l'algorithme RSA

Chiffrement par l'algorithme RSA

```
void __fastcall TMainForm::RSA1Click(TObject *Sender)
{
// chiffrement RSA
Form4->Label4->Caption="";
Form4->Label4->Caption="التشفير بال RSA";
Form4->ShowModal();
/*Texte vide*/
int S = RichEdit1->GetTextLen();
if((S==0)&&(Accepter_rsa)){
         Accepter_rsa=false;
         ShowMessage("لا يمكن تشفير نص فارغ");
         }
/*phase de chiffrement*/
if (Accepter_rsa){
int star=clock();
int i,j,k;
int Size = RichEdit1->GetTextLen();

char *Buffer = new char[Size+1];
RichEdit1->GetTextBuf(Buffer,Size+1);
RichEdit1->Text="";
unchar t1[8], t2[8];

ProgressBar1->Visible = true;
ProgressBar1->Min = 0;

unsigned long e,n;
e =(clef1);
n =(clef2);
```

```
int n56=Size/7;
if (Size % 7 != 0) n56 = (n56+1)*7;
        else n56 = Size;

ProgressBar1->Max = n56;

unchar *tab1 = new unchar[n56+1];
unchar *tab2 = new unchar[n56+1];

for (i=0;i<n56;i++)  tab1[i]= table_rsa[114].entier;

for (i=0;i<Size;i++){
           j=0;
           while (Buffer[i]!=table_rsa[j].lettre && j++<159);
           tab1[i]=table_rsa[j].entier;
};
delete Buffer;
ProgressBar1->Position=1;
// Chiffrement
for(i=0;i<n56;i++)
    tab2[i] = modexp(tab1[i], e, n);
delete tab1;
ProgressBar1->Position=2;
// fin du chiffrement

// pour eviter un eventuel pb d'affichage.
int som;
som = n56/7;
som++;
unchar *TabEtend = new unchar[n56+som+1];
int enc=0;
i=0;k=0;
```

```
while(i<n56){
    for(j=0;j<7;j++,i++)  t1[j]= tab2[i];//le tableau contenat les ent chiffres
    de7vers8(t1,t2);enc++;
    for(j=0;j<8;j++,k++) TabEtend[k] = t2[j];
};
delete tab2;
ProgressBar1->Position=3;
//////
char **les_cars;

les_cars = new char*[1];
    les_cars[0] = new char[n56+enc+1];

for (i=0;i<n56+enc;i++){
       j=0;
       while((TabEtend[i]!=table_rsa[j].entier) && j++<159);
       les_cars[0][i]= table_rsa[j].lettre;
};
les_cars[0][n56+enc]='\0';

for (i=0;i<n56+enc;i++){
if(i%5==0) ProgressBar1->Position=i;
RichEdit1->Text = RichEdit1->Text + les_cars[0][i];}
////////
desallouer_rsa(les_cars);
delete TabEtend;
////////
ProgressBar1->Visible=false;
int end=clock();
Label2->Caption="";
Label2->Caption=((end-star)*0.001);
}
}
```

Déchiffrement par l'algorithme RSA

```
void __fastcall TMainForm::RSA2Click(TObject *Sender)
{

//déchiffrement RSA
Form4->Label4->Caption="";
Form4->Label4->Caption="فك التشفير بال RSA";
Form4->ShowModal();

int S = RichEdit1->GetTextLen();
if((S==0)&&(Accepter_rsa)){
         Accepter_rsa=false;
         ShowMessage("لا يمكن فك تشفير نص فارغ");};

if (Accepter_rsa){
int star=clock();
int i,j,k;
int Size2 = RichEdit1->GetTextLen();
unsigned long e,n;
e = (clef1);
n = (clef2);

char *Buffer2 = new char[Size2+1];
RichEdit1->GetTextBuf(Buffer2,Size2+1);
RichEdit1->Text="";
unchar t3[8],t4[8];

ProgressBar1->Visible = true;
ProgressBar1->Min = 0;
```

```
int n64=Size2/8;
if (Size2 % 8 != 0) n64 = (n64)*8;
        else n64 = Size2;

ProgressBar1->Max = n64;
int som2;
som2 = Size2/8;

unchar *tab2 = new unchar[Size2+1];
for (i=0;i<Size2;i++)  tab2[i]= table_rsa[114].entier;

for (i=0;i<Size2;i++){
           j=0;
           while (Buffer2[i]!=table_rsa[j].lettre && j++<159);
           tab2[i]=table_rsa[j].entier;
};
delete Buffer2;
ProgressBar1->Position=1;
//maintenant intervient ma fonction avancee
unchar *TabRes = new unchar[Size2+1];
i=0;k=0;
while(i<Size2)
{
    for(j=0;j<8;j++,i++)  t3[j]= tab2[i];
    de8vers7(t3,t4);
    for(j=0;j<7;j++,k++) TabRes[k] = t4[j];
};
// Déchiffrement
ProgressBar1->Position=2;
unchar *tab3 = new unchar[k+1];
```

```
for(i=0;i<k;i++)
    tab3[i] = modexp(TabRes[i], e, n);
delete tab2;
ProgressBar1->Position=3;
// fin du chiffrement

for( i=0; i<k; i++)
        tab3[i] =(unchar)(tab3[i]&0x7f);
char **les_cars2;

les_cars2 = new char*[1];
    les_cars2[0] = new char[Size2-som2+1];
//char *les_cars2 = new char[Size2-som2+1];
for (i=0;i<Size2-som2;i++){
       j=0;
       while((tab3[i]!=table_rsa[j].entier) && j++<159);
       les_cars2[0][i]= table_rsa[j].lettre;
};
les_cars2[0][Size2-som2]='\0';
//RichEdit1->Text = les_cars2;
for (i=0;i<Size2-som2;i++){
if (i%5==0) ProgressBar1->Position=i;
RichEdit1->Text = RichEdit1->Text + les_cars2[0][i];}
desallouer_rsa(les_cars2);
delete tab3;
ProgressBar1->Visible=false;
int end=clock();
Label2->Caption="";
Label2->Caption=((end-star)*0.001);}}
```

4.9. Conclusion

Nous sommes intéressé dans notre projet de fin d'étude par le cryptage des lettres arabes. Il existe deux classes d'algorithmes de cryptages dans le domaine de la cryptographie.

Notre objectif est basé sur le cryptage des textes arabe par l'algorithme de chiffrement à clé secrète DES et l'algorithme de chiffrement à clé publique RSA.

Nous souhaitons dans les prochains projets de continué dans ce domaine mai en utilisons d'autre algorithmes de chiffrements standard plus puissante et reconnu dans le monde comme AES et Diffi HELMAN.