オブジェクト指向とゲームプログラミング

● 基礎編 ― 第16回　暗号 ●

暗号

暗号とは、機密文書などの重要なデータが漏れても、第三者にわからないようにする仕組みのことです。暗号化されていないデータを平文と呼びます。平文を暗号化すると暗号文になります。暗号文は、復号すると平文に戻ります。第三者が暗号の仕組みを解析することを解読といいます。

代表的な暗号化の方法に、アルファベット順にずらすというシーザー暗号があります。たとえば、"IBM"を１つ前にずらすと"HAL"に、７つ後にずらすと"PIT"になります。復号には、何文字ずらしたかの情報が必要になります。このような復号のための補助情報を暗号の鍵(キー)といいます。

シーザー暗号と似たものに、ビットごとの排他的論理和による暗号化があります。ビットごとの排他的論理和は、２回行うと元に戻ります(a xor b xor b = a)。したがって、暗号文を同じプログラムにもう一度とおせば平文に戻ります。この方法では、データは一見でたらめなバイト列と化します。しかし、(暗号鍵が８ビットの場合)256通りの鍵をすべて試すだけで解読できます。鍵を数バイト長にしても、解読はあまり難しくなりません。解読を難しくするには、１バイトごとに鍵を乱数で変えるなどの方法をとります。暗号化の前に、データ圧縮を行うと解読が難しくなります。

本格的な暗号としては、IBMが開発したDES(Data Encyption Standard)やマサチューセッツ工科大学のRivest, Shamir, Adlemanが開発したRSAが有名です。

以下のプログラムは、排他的論理和と乱数を組み合わせた簡単な暗号化のプログラムです。暗号化データを再度このプログラムにかけると元のデータに戻ります。

```
const int   KEY = 1234;                     // 暗号鍵

char   in_str [1024];                       // 入力データ
char   out_str[1024];                       // 出力データ

srand(KEY);                                 // 乱数初期化

int   i;
for(i = 0; i < strlen(in_str); i++) {
    const int   r = rand() / 256;           // 0～255の乱数を生成
    out_str[i] = in_str[i] ^ r;             // 乱数と排他的論理和をとって暗号化する
}
out_str[i] = '\0';
```

暗号化は、ゲームでもよく使用されます。たとえば、アドベンチャーゲームです。会話などのデータは、そのまま記録しておくと、ゲームをプレイしなくてもストーリーがわかってしまいます。そのためほとんどのゲームでは、暗号化して記録しておき、必要なときに復号して使用しています。

```
[              ;autoload =cont <cont  2cont = <  `          X          [c ・      ;new [          ;new [c         ;n
ew X          ;ara 2new X                    .bg\hai_21 ^p

 t Lpトbャp喘鈎・ノb澱B"     Lv甲ゃp屠Mbｲcjc9c`b･a!c^a;csc^a;csbヲv「bC､bれbノa"     LjZbゃj枌ャkｪbミubb叡ｷbｲb鈎ヌa!c
<a;c?v｜ャc!c#c8歯ミc+cjcBbヲb叡肪ノa"     KN LaUd･d･b「bヂ!s鈎ゃb秒ｽbノaV     Lb礎ソb叡ｽbミc2csc2csbヲr b誼ｲb鈎ヌa!kモb
ミk「b斬"      .bg\white
  .bg\img_001
     Lb肪ネb肪ネbヲoヲbネb「b獣ノq檜Zb蛎!p>oテbゃbあ｡b拌"歯礎ハbｪokbCｪb話枌ヲa!t-b糞ヌbハb拯ut獣ゃnネb暫母bミrｪb預"
 KN LaUbC叡ゃbｦi､b｡b､j巴肪枌蛎!bシb枴､bヌb秒C礎ヲ鸞 bｪbハb「bｸb秒､bｲbCｲaV     Lijs6i喘･c!c#c8c,cccscFc"a;bミb叡
肪ノbャbｪa!i､歯ヌb秒L#b鈎傭ノbャb枌蛎!i巴鈎C・､bヌb秒C礎ヲ歯ヲbC・ャbｪa!
 Lbツbｪbネa!j#pewabヲb餌!i､mオbネbヲb餌!b嫦・C・#鹹XcscGbャb礎ヲb杪"
 KN LaUbシa!m《ﾚbｪi況鍛殻!b｡b､b甲ノb｡b､bC・病V     Lhハiｰa!bあC「bャb拇ソbゃnニb｡b､b・C拇ヲbC・ョa!黏傭礎・鈎CC
枌ﾋb・"     Li況鍛ャrｪb｡b､bCノj･l7jWm:bｪa!uwl泳Wi`bヲuゃbへ歯ヌbｲbJ､kXbｽbｲbャb枌蛎!bツbノb礎ヲb鈎ゐノb叡ヌbヲa!麕!oｲb較
Xbハb､bC拌"     Lk丁\c0cBcGbャrfb･lGq7b蛎!lYb膏Kb・拌"     .bg\hai_21 ^p
   KN LaUcaa;ckaCaCi況鍛叡ヌb餌V   M  Li況鍛ャoニbノc0c8bャn~jｴb覚qbC､bあ｡b拌"b「bシbネ歯ｪl]b・ゃwhbCヲbC・礎ヲb餌"
    Li巴ｪa!uｴnvbタq･b拇話ゃa!lGqアbミc\c0cBcGbゃb秒シ歯C礎ムb秒"     KN LaUbシbあ!i況鍛殻!b｡b､b甲ノbｲbヌa!bニb・ツb絞巴
ゃ鸞 bタi､kXbｽb革+bハb拇｡b､b礎ヲbゃbｲbノbｲaV     Lb嫦・ヲlhbシbハbヨd･d･a!   2sel1     E01 Ma{i況鍛ミl]b・ゃmSb礎・) E02
 Ma{lnoノs bゃw6bアbゃmSb礎・)  M 201 [
```

▲暗号化されたシナリオファイル

練習問題

1　２５６文字格納できる文字配列plain_strを宣言し、適当な文字列で初期化しましょう。

2　２５６文字格納できる文字配列cipher_strを宣言しましょう。

3　２５６文字格納できる文字配列decrypt_strを宣言しましょう。

4　文字配列plain_strの文字列をアルファベット順に１つ前にずらして暗号化し、文字配列cipher_strに格納しましょう。

5　文字配列cipher_strの文字列を復号して文字配列decrypt_strに格納しましょう。

6　文字配列plain_str, cipher_str, decrypt_strを画面に表示しましょう。

7　文字配列plain_strの文字列をアルファベット順に７つ後にずらして暗号化し、文字配列cipher_strに格納しましょう。

8　文字配列cipher_strの文字列を復号して文字配列decrypt_strに格納しましょう。

9　文字配列plain_str, cipher_str, decrypt_strを画面に表示しましょう。

10　0～255の任意の数値を暗号鍵とし、その数値と文字配列plain_strの文字と排他的論理和をとって暗号化し、文字配列cipher_strに格納しましょう。

11　文字配列cipher_strの文字列を復号して文字配列decrypt_strに格納しましょう。

12　文字配列plain_str, cipher_str, decrypt_strを画面に表示しましょう。

13　以下のようなテキストファイルを作成し、"plain.txt"という名前で保存しましょう。

```
'----------------------------------------
#通学途中
'----------------------------------------
#BG1 Town.bmp
#HITO d.bmp
俺は王魔。@自分で言うのも何だが…@九十九高校に通うちょっと変わった@高校２年生だ。
何が変わってるかって？@……そう、俺は自他ともに@認める女の子好きなのさ。
@だが、それはもう過去の話。@今俺は真のパートナーを@決めようとしているんだ。
で、その最終候補に残ったコが３人@いるんだ。
```

14　以下のプログラムを参考に、13のファイルを適切な方法で暗号化し、"cipher.txt"というファイルに出力しましょう。

```
FILE*   infile  = fopen("入力ファイル名", "rb");
FILE*   outfile = fopen("出力ファイル名", "wb");

if(infile == NULL || outfile == NULL) {
    puts("入力ファイルまたは出力ファイルが開けません");
    return EXIT_FAILURE;
}

int   c = fgetc(infile);    // 入力ファイルから１バイト読み込む
while(c != EOF) {           // 入力ファイルの終わりまでループ
    c = (暗号化処理);       // ※ ここに、暗号化処理を記述 ※
    fputc(c, outfile);      // 暗号化したデータを出力ファイルに書き出す
    c = fgetc(infile);      // 次のデータを入力ファイルから読み込む
}
```

15　14のファイルを復号し、"decrypt.txt"というファイルで出力しましょう。