「暗号解読に挑戦」–東海大学情報通信学部–

暗号解読プログラムは「yamamoto hiroshi」でウェブ検索し，「Yamamoto Hiroshi (Educational contents) - Yamamoto Lab.」を開いて「暗号解読デモ」のリンクにある「暗号解読デモプログラム」を開いてください．

1　キーの長さを決める

上記「暗号解読に挑戦」ページの「暗号解読デモプログラム」を開いて最初に表示されている，0,1 が連続したものが暗号文です．0 か 1 一つを「1 ビット」と言います．画面をよく見ると 8 ビットずつ区切られていることがわかります．コンピュータでは多くの場合 8 ビットを一つのデータのまとまりとして処理します．

この暗号は 8 ビットのデータ数個をキーとして使用します．この個数を「キーの長さ」といいます．キーは前もって送受信者が相談して決めておき，当事者以外には秘密にしておきます．もとの文（平文-ひらぶん-といいます）を暗号文にに変えるためにキーを使い，逆に暗号文からキーを使って平文に戻します．このキーを第三者（あなた）が見破れば暗号を破ることができます．

暗号文の下に今回の暗号文の統計データが表示されています．暗号文は 8 ビットのデータの連続ですが，暗号解読の最初の手がかりとして暗号文とそれを何文字かずらしたものを並べ，一致する確率を調べます．例えば「ずれ量」1 の「一致確率」は暗号文とそれ自身を 1 文字先にずらしたものが一致する確率を計算したものです．

00010100		↓ずれ量 1
01011011	··· 一致？ ···	00010100
00010101	··· 一致？ ···	01011011
00110011	··· 一致？ ···	00010101
⋮		⋮

00010100		
01011001		↓ずれ量 2
00010101	··· 一致？ ···	00010100
00110011	··· 一致？ ···	01011011
⋮		⋮

先ほどの「一致確率」の項目を見ると，ある周期で確率が高くなっているはずです．その周期がキーの長さになります．例えば以下のデータは，ずれ量 3,6,9 の確率が他と比較して高いのでキーの長さは “3” になります．数字ではなく桁数に注目します．

ずれ量	一致確率
1	0.00357568533969011
2	0.0035799522673031
3	0.0884109916367981
4	0.00478468899521531
5	0.00239520958083832
6	0.0551558752997602
7	0.00120048019207683
8	0.00360576923076923
9	0.075812274368231

このプログラムではキーの長さは 3 から 5 の範囲でランダムに選ばれます．

作業 1：「一致確率」がどの周期で大きくなっているかを見極め，自分が推定したキーの長さを「キーの長さ」欄に入力し，「送信」ボタンを押して下さい．

2　各キーを決める

キーの長さが決まればそれぞれ個別のキー文字の 8 ビットが何なのかを決めます．これができればキーを完全に見破ったことになり，暗号を破ったことになります．（どんな暗号文でも平文に戻すことができる）

例えばキーの長さが 3 の場合でキー文字を最初の文字から順にキー 1，キー 2，キー 3，だったとします（それぞれ 8 ビット）．暗号文はキーを繰り返し，すなわちキー 1，キー 2，キー 3，キー 1，キー 2，キー 3，キー

1，キー 2，キー 3. . . と並べたものを平文と XOR と呼ばれる演算 “⊕” を行って作られます．

平文 1	平文 2	平文 3	平文 4	平文 5	平文 6	平文 7	平文 8	平文 9
⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕
キー 1	キー 2	キー 3	キー 1	キー 2	キー 3	キー 1	キー 2	キー 3
||	||	||	||	||	||	||	||	||
暗号文 1	暗号文 2	暗号文 3	暗号文 4	暗号文 5	暗号文 6	暗号文 7	暗号文 8	暗号文 9

キーの長さが 3 の場合，キーは 3 個しかありません．1, 4, 7, . . . 文字めの暗号文は キー 1 で，2, 5, 8, . . . 文字めの暗号文は キー 2 で，3, 6, 9, . . . 文字めの暗号文は キー 3 で変換されています．

キーの長さが 3 の場合，画面の「出現確率」には 1 から 3 までのキー位置の暗号文の統計情報が表示されます．キー位置 1 は暗号文の $1, 4, 7, \ldots$ 番目の文字についての統計，キー位置 2 は暗号文の $2, 5, 8, \ldots$ 番目の文字についての統計，キー位置 3 は暗号文の $3, 6, 9, \ldots$ 番目の文字についての統計です．同じキー位置の文字はすべて同じキーで暗号化されています．「1 位」にそれぞれのキー位置で最も出現頻度の高かった暗号文の文字とその頻度が表示されています．

作業 2：下の「1 位」の各欄に各キー位置の「1 位」 に表示された暗号文の 8 ビットを記入して下さい．（今は 2 位以下は使用しません）キーの長さが 3 の場合はキー位置 4 以降は空欄に，キーの長さが 4 の場合はキー位置 5 以降は空欄にして下さい．

キー位置	1	2	3	4	5
1 位					
	⊕	⊕	⊕	⊕	⊕
空白記号	0 0 1 0 0 0 0 0	0 0 1 0 0 0 0 0	0 0 1 0 0 0 0 0	0 0 1 0 0 0 0 0	0 0 1 0 0 0 0 0
	||	||	||	||	||
キー					

次に必要になるのが英語の各文字の出現頻度です．空白文字を含んだ英文テキストの場合，空白文字が最も出現頻度が高いことが知られています．このことから，各ポジションの「1 位」の 8 ビットが元の平文では空白記号であったと予想します．

空白記号に対応する 8 ビットは ASCII（アスキー）と呼ばれる規格で”001000000” と決められています．

まず，キー 1 について説明します．空白記号 “00100000” とキー 1 で ⊕ 演算を行ったものが 上で記入した「キー位置 1」 の 「1 位」になるように「キー 1 」を推測します．具体的には以下のようにしてキーを決定します．

作業 3：キー位置 1 から順に，上で「1 位」欄に記入したビット列の左から 3 桁目だけを反転（0 → 1, 1 → 0）させたものを「キー」欄に記入して下さい．3 桁目以外は「1 位」欄のものをそのまま写して下さい．どのキーの長さ，キー位置でも反転させるのは 3 桁目だけです．

作業 4：作業 3 で記入したキーすべてを「キー」欄に入力して「送信」ボタンを押して下さい．

自動的に正しいかどうかチェックされます．正しければ「おめでとうございます」の表示と暗号化される前の平文が，間違っていれば「キーが正しくありません」の表示と，指定したキーで復号を試みたものが表示されます．

統計を利用した手法なので手順通りに行っても必ず正しく解読できるとは限りません．推測したキーが正しくなかった場合で推定したキーの一部を修正したい場合は，ブラウザの「戻る」ボタンで戻って入力をやりなおすことができます．推測が間違っていそうなキーについて 1 位ではなく 2 位が空白文字だったと仮定してキーを計算し，やりなおしてみて下さい．問題の選択からやり直したい場合は 「最初からやりなおす」を押して下さい．