1. Режим лінійної зміни частот символів (“barrier”)

1А. Підрежим для пропуску: $f\_$ = fixed const = f_0 (частота пропуску фіксована на інтерфейсі)

Вводимо з інтерфейсу значення параметрів $f\_ = f_0$, M, C.

1. Частоти символів задають залежно від їхнього рангу:

$$f_1 = f_1,$$
$$f_2 = f_1 - a,$$
$$f_3 = f_1 - 2a,$$
$$\ldots$$
$$f_M = f_1 - (M-1)a,$$

де i – це ранг символу (i = 1, 2, …, M), f_1 – шукана найвища частота деякого символу (ранг цього символу i = 1), M – алфавіт символів (без пропуску!), який задають з інтерфейсу, a – шукана константа. Відомо, що максимальний перепад частот тут дорівнює

$$f_1 / f_M = C,$$

де параметр C задають з інтерфейсу (це т.зв. «бар’єр»).

2. Формули для параметрів f_1 і a:

$$f_1 = \frac{2C(1-f_0)}{M(C+1)},$$
$$a = \frac{C-1}{C+1}\,\frac{2(1-f_0)}{M(M-1)}.$$

Ці формули я одержав з умови нормування суми частот (тобто ймовірностей) символів і умови лінійної зміни частоти символів залежно від їхнього рангу.

1Б. Підрежим для пропуску: $f\_ = f_1$ (частота пропуску не фіксована, але вона завжди найвища, тобто перша за рангом, тобто для пропуску ранг дорівнює i = 1)

Вводимо з інтерфейсу значення параметрів M, C.

1. Частоти символів:

$$f_1 = f_1,$$
$$f_2 = f_1 - a,$$
$$f_3 = f_1 - 2a,$$
$$\ldots$$
$$f_M = f_1 - (M-1)a,$$
$$f_{M+1} = f_1 - Ma,$$

де i – це ранг символу (i = 1, 2, …, (M + 1)), f_1 – шукана частота символу з найвищою частотою (це пропуск!), (M + 1) – розширений «алфавіт» символів (разом із пропуском!), a – шукана константа. Тут уже максимальний перепад частот визначають як

$$f_1 / f_{M+1} = C,$$

де параметр C також задають з інтерфейсу (це знову т.зв. «бар’єр»).

2. Формули для параметрів f_1 і a:

$$f_1 = \frac{2C}{(M+1)(C+1)},$$

$$a = \frac{C-1}{C+1} \, \frac{2}{(M+1)M} \, .$$

2. Режим логарифмічної зміни частот символів (“log”)

Вводимо з інтерфейсу значення параметрів $f\_ = f_0$, M, A.

Частоти символів шукають за формулою

$$f_i = A - B\lg i \, ,$$

де i – це ранги символів, параметр A задають з інтерфейсу (але уважно! – див. далі), а параметр B не вводять з інтерфейсу, а розраховують! Як і вище, це також роблять з умови нормування частот символів.

2А. Підрежим для пропуску: $f\_$ = const = f_0 (частота пропуску фіксована на інтерфейсі)

Вводимо з інтерфейсу значення параметрів $f\_ = f_0$ і M.

Далі вводимо деяке **додатне** значення параметра A, обраного нами для заданих f_0 і M.

Перевіряємо такі умови:

$MA > 1 - f_0$ (умова додатності параметра B – див. нижче), (1)

$A < \dfrac{1 - f_0}{M - \dfrac{\lg(M!)}{\lg M}}$ (умова додатності всіх частот f_i, у т.ч. найнижчої, при $i = M$), (2)

де M – алфавіт символів (без пропуску!).

Якщо умова (1) не виконується, то слідує оголошення «Надто малий параметр A!». Тоді треба ввести з інтерфейсу більший параметр A.

Якщо умова (2) не виконується, то слідує оголошення «Надто великий параметр A!». Тоді треба ввести з інтерфейсу менший параметр A.

Зауважимо, що ще одна тривіальна умова, накладена на параметр A з міркувань, аби принаймні перша (найвища) частота f_1 не перевищувала відведену на символи суми частот (тобто умова $f_1 = A < 1 - f_0$), слабша за умову (2), а тому її не потрібно враховувати. Це випливає хоча би з того, що знаменник справа у формулі (2) завжди більший за одиницю, якщо тільки $M > 1$.

Якщо обидві умови (1) і (2) виконуються, то приймаємо введене з інтерфейсу значення A.

Далі з умови $\sum_i f_i = 1 - f_0$ визначаємо параметр B:

$$B = \frac{MA - (1 - f_0)}{\lg(M!)} \, .$$

2Б. Підрежим для пропуску: $f\_ = f_1$ (частота пропуску не фіксована, але вона завжди найвища, тобто перша за рангом, тобто для пропуску ранг $i = 1$)

Вводимо з інтерфейсу значення параметра M.

Для заданого M вводимо значення A.

Перевіряємо такі умови:

$$(M+1)A > 1, \tag{1’}$$

$$A < \frac{1}{(M+1) - \dfrac{\lg[(M+1)!]}{\lg(M+1)}} \, , \tag{2’}$$

де $(M + 1)$ – розширений «алфавіт» символів (разом із пропуском!).

Якщо умова (1’) не виконується, то слідує оголошення «Надто малий параметр А!». Тоді треба ввести з інтерфейсу більший параметр A.

Якщо умова (2’) не виконується, то слідує оголошення «Надто великий параметр А!». Тоді треба ввести з інтерфейсу менший параметр A.

Якщо умови (1’) і (2’) виконуються, то приймаємо введене з інтерфейсу значення A.

Далі визначаємо такий параметр B:

$$B = \frac{(M+1)A - 1}{\lg[(M+1)!]}.$$

Повторно зазначимо: тут частота пропуску завжди найвища, тобто $f_1 = f\_ = A$ (ранг пропуску $i = 1$)!

Зауваження:

1) у формулах із $\lg(M!)$ або $\lg[(M+1)!]$ слід зробити такі заміни:

$$\lg(M!) = \lg 2 + \lg 3 + \ldots + \lg M,$$

$$\lg[(M+1)!] = \lg 2 + \lg 3 + \ldots + \lg M + \lg(M+1).$$

Правда, такі формули «неаналітичні» (через «три крапки» ☺) і їх треба програмувати накопиченням у циклі, але зате швидкодія їхнього розрахунку на комп’ютері набагато вища!

2) хоча зараз це непотрібно і навіть шкідливо для швидкодії програми, проте в інших ситуаціях у програмуванні розрахунку логарифму факторіалу числа можна використовувати формулу Стірлінґа

$$\lg(n!) \approx \lg\left[\sqrt{2\pi n}\left(\frac{n}{e}\right)^n \left(1 + \frac{1}{12n} + \frac{1}{288n^2} - \frac{139}{51840n^3} + \ldots\right)\right],$$

де $n = M$ або $n = (M + 1)$ відповідно в підрежимах (А) або (Б).

Можливо, за умови (наприклад) $n > 100$ слід було би обрізати кількість членів у формулі Стірлінґа – хоча малоймовірно, що це істотно додасть до швидкодії, а тому чи вартує це робити?..

Природно, що в формулі логарифм добутку слід замінити на суму логарифмів, логарифм степеня числа на добуток цих степеня та числа – для спрощення розрахунків на компі.

3) у старій програмі М. Альфавіцького на C# логарифмічний режим уведено непослідовно! А саме, він більш чи менш коректно (хоча не ідеально!) працює лише для випадку максимального алфавіту $M = 26$. Насправді програма тоді «трохи ігнорує» задані A і B і обирає натомість відносно близькі, але не саме такі значення A і B. Виходить логарифмічний режим частот символів, проте з дещо іншими константами. Це перевірено! І нащо таке робити?!

А от при спробі ввести довільний (менший) алфавіт ($M < 26$) програма виводить на інтерфейсі неправильні частоти (сума яких не нормується на одиницю або на величину $(1 - f_0)$), а вихідний текст і надалі має алфавіт $M = 26$ ☺. Одна з причин цього: не можна наперед вводити обидва параметри логарифмічного закону A і B, ще не знаючи точного розміру алфавіту! Ці параметри не можна вважати незалежними! Один із цих параметрів таки можна ввести (найкраще A), а от параметр B залежатиме від розміру алфавіту.