Bonus Compression Elias-Gamma

Coding γ untuk suatu bilangan bulat positif k terdiri dari dua komponen:

- Sektor;
- Body;

Untuk mengencode:

- Cari nilai terbesar dari N, sehingga $2^N \leq X$, dengan kata lain, kita mencari Most significant bit dari X, kemudian kita akan melakukan encoding bilangan N menggunakan unary coding, itu untuk bagian sectornya. Unary coding untuk suatu bilangan N merupakan $N-1$ buah 0 diikuti dengan satu buah bit 1.
- Bagian sisanya, yaitu $X - 2^N$, direpresentasikan dengan binary, yaitu bagian bodynya.
- Bagian sector pada intinya sama seperti penunjuk size dari bilangannya, sehingga nanti cara bacanya tinggal
- kira-kira:

```
while(*bit != 1) len++
bilangan++
while(len--) bilangan <<= 1, bilangan += bit
```

Misal bilangan $7 = 2^2 + 3 = 00111$

Integer	Binary	Gamma Code	Delta Code	Fibonacci Code
1	1	1	1	11
2	10	010	0100	011
3	11	011	0101	0011
4	100	00100	01100	1011
5	101	00101	01101	00011
6	110	00110	01110	10011
7	111	00111	01111	01011
8	1000	0001000	00100000	000011
9	1001	0001001	00100001	100011
10	1010	0001010	00100010	010011

3.   Average Encoding (AE)

Untuk benchmarking didapatkan hasil:

```
index file size: 49724 Bytes = 48.55859375 KB
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 55.26it/s]
Average time searching for VBEPostings: 0.45737 seconds
(env) hocky:~/kuliah/ir/TP1$ python3 benchmark.py
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 21.48it/s]
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 55.07it/s]
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 51.16it/s]
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 60.45it/s]
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 60.19it/s]
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 57.03it/s]
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 55.67it/s]
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 60.49it/s]
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 60.58it/s]
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 51.13it/s]
Average time indexing for VBEPostings: 0.47410 seconds
dict file size: 67599 Bytes = 66.0146484375 KB
index file size: 49724 Bytes = 48.55859375 KB
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 60.62it/s]
Average time searching for VBEPostings: 0.48441 seconds
(env) hocky:~/kuliah/ir/TP1$ python3 benchmark.py
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 21.62it/s]
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 65.41it/s]
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 65.56it/s]
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 65.40it/s]
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 65.33it/s]
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 65.37it/s]
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 65.62it/s]
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 65.41it/s]
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 65.20it/s]
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 65.54it/s]
Average time indexing for StandardPostings: 0.39821 seconds
dict file size: 76809 Bytes = 75.0087890625 KB
index file size: 348680 Bytes = 340.5078125 KB
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 65.56it/s]
Average time searching for StandardPostings: 0.40845 seconds
(env) hocky:~/kuliah/ir/TP1$ python3 benchmark.py
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 20.98it/s]
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 52.91it/s]
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 53.06it/s]
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 52.97it/s]
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 52.85it/s]
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 52.96it/s]
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 52.85it/s]
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 52.83it/s]
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 52.63it/s]
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 52.83it/s]
Average time indexing for EliasGammaPostings: 0.57993 seconds
dict file size: 67581 Bytes = 65.9970703125 KB
index file size: 42705 Bytes = 41.7041015625 KB
100%|████████████████████████████████████████████| 20/20 [00:00<00:00, 51.78it/s]A
verage time searching for EliasGammaPostings: 0.59071 seconds
(env) hocky:~/kuliah/ir/TP1$ git add .
```

Compression Method	Waktu Rata-Rata Indexing	Waktu Rata-Rata Searching	Ukuran Dictionary	Ukuran Index
Standard (Tidak ada kompresi)	0.39821 detik	0.40845	75 KB	340.5 KB
Variable Byte Encoding	0.47410 detik	0.48441 detik	66.01 KB	48.56 KB
Elias Gamma	0.57993 detik	0.59071 detik	66 KB	41.704 KB

Dari sini kita mengetahui bahwa terdapat trade off waktu dan memori dari beberapa teknik kompresi yang dilakukan, dengan pencarian dilakukan 10 kali untuk query:

```
queries = ["olahraga", "tumor", "hidup sehat", "jantung", "penyakit",
            "kuat", "badan sehat", "olahraga teratur", "tidur cukup"]
```

Serta dilakukan indexing 10 kali pula untuk masing-masing teknik kompresi.