

## Zadanie J\*

### Kompresja pliku

Centralna Agencja Wywiadowcza Bajtolandii postanowiła zmienić sposoby przekazywania tajnych informacji między agentami. Aby zwiększyć bezpieczeństwo przekazywanych danych agenci będą się posługiwać nieużywanymi już w obiegu dyskietkami 3,5-calowe HD o pojemności 1,44MB. W razie przechwycenia dyskietki przez niepowołane ręce odczyt danych z dyskietki będzie utrudniony ze względu na brak stacji dysków dla wspomnianych dyskietek we współczesnych komputerach. Ze względu małą pojemność dyskietek, przekazywane dane będą skompresowane. Zwiększy to dodatkowo ich bezpieczeństwo.

Twoim zadaniem jest napisanie programu, który będzie kompresował i dekompresował zadane pliki tekstowe. Pliki te będą się składać z 26 małych liter alfabetu angielskiego (a-z), spacji, kropki oraz znaków enter. Każdemu znakowi przypisany jest kod - ciąg bitów. Ciągi te są różnej długości i tworzą kod prefiksowy, tzn. żaden kod nie jest prefiksem kodu przypisanemu innemu znakowi. Kompresja pliku polega na zastąpieniu każdego znaku (ciągu 8-bitowego) przez jego kilku bitowy kod. W ten sposób powstanie ciąg bitów. Każde 8-bitów powstałego ciągu jest interpretowanych jako znak, zapisywany do wynikowego pliku binarnego. W przypadku, gdy długość wynikowego ciągu bitów nie jest podzielna przez 8, jest on na końcu uzupełniany zerami.

Kodowanie znaków:

a	1110	b	1000111
c	10111	d	10000111
e	1001	f	1000100
g	1000110	h	1000101
i	1100	j	111110
k	00011	l	1000001
m	1000000	n	0101
o	1010	p	00010
q	100001001	r	0100
s	11011	t	11010
u	111111	v	100001000
w	11110	x	10000101
y	10110	z	0010
spacja	0111	kropka	0110
enter	0011		

Program powinien być uruchamiany **z trzema parametrami**: dwoma nazwami plików: `plik.in` `plik.out` oraz liczbą  $a$  ( $a = 1$  lub  $a = 2$ ). Jeśli  $a = 1$  program kompresuje plik `plik.in` i zapisuje wynik do pliku `plik.out`, jeśli  $a = 2$  program rozpakowuje plik `plik.in` uzyskując oryginalny plik i zapisuje go do pliku `plik.out`. Pliki przygotowane są w systemie Linux.

Wywołanie programu:

```
./program plik.in plik.out a
```

Dostępna pamięć: 2MB

Wymagany język: C

## Przykład

Dla wywołania: `./program plik.in plik.out 1`

Plik `plik.in`:

```
 pewnego dnia puchatek wyszedl na spacer. az zaszedl na polane w srodku lasu.  
a posrodku tej polanki rosl wielki dab  
i z samego jego wierzcholka dochodzilo glosne bzykanie.
```

```
kubus puchatek usiadl sobie pod tym debem.  
podparl glowe na lapkach i zaczal rozmyslac.
```

Binarny plik `plik.out` składa się z ciągu bitów, których wartości liczbowe są równe kolejno:

```
20 249 102 53 60 58 231 56 191 188 94 212 141 253 109 148 195 193 117 231 216  
186 242 140 252 78 93 178 152 120 46 188 226 168 60 178 254 125 165 67 143 247  
131 219 253 143 156 85 180 168 113 254 250 159 156 85 7 148 120 233 91 130 254  
201 130 60 120 126 142 120 228 251 232 19 26 159 233 141 79 236 148 43 197 168  
35 231 135 171 197 168 114 200 52 241 160 214 214 94 57 88 124 185 44 102 63 227  
255 183 23 247 139 218 145 191 251 206 135 130 251 168 249 46 42 135 125 90 3  
195 204 121 128 198 42 135 23 36 23 141 6 189 46 188 240 120 67 235 197 124 114  
235 151 65 116 162 129 109 193 235 177 128
```