Politechnika Wrocławska, Informatyka Stosowana

Kryptografia Historyczna

Cyberbezpieczeństwo, Laboratorium nr.2 - raport

Autor: Aleksander Stepaniuk
Nr. Indeksu: 272644

Zad 1. Algorytmy historyczne – właściwości.

Teksty których użyłem do analizy kolejnych algorytmów:

Tekst 1:

„Jeszcze gdy chodzilem do podstawowki to byl tam taki Pawel i ja jechalem na rowerze i go spotkalem i potem jeszcze pojechalem do biedronki na lody i po drodze do domu wtedy jeszcze juz do domu pojechałem”

Tekst 2:

„Cesarz czesal wlosy cesarzowej cesarzowa czesala wlosy cesarza Dzdzystym rankiem gzegzolki i piegze zamiast wziac sie za dzdzownice nazarly sie na czczo miazszu rzezuchy i rzedem rzygaly do rozzarzonej brytfanny Idzie Sasza sucha szosa suszy sobie swoje szorty Gdzie jest kufel pyta brat Moze kufel w kufer wpadł Bracie zawsze ci tlumacze kufel wpadl do kufra raczej Wyjmij z kufra kufel bracie lepiej postaw go na blacie”

Tekst 3:

„AAAAAABBBBCCCCCCCCCCCCCCDDDDDDDDD”

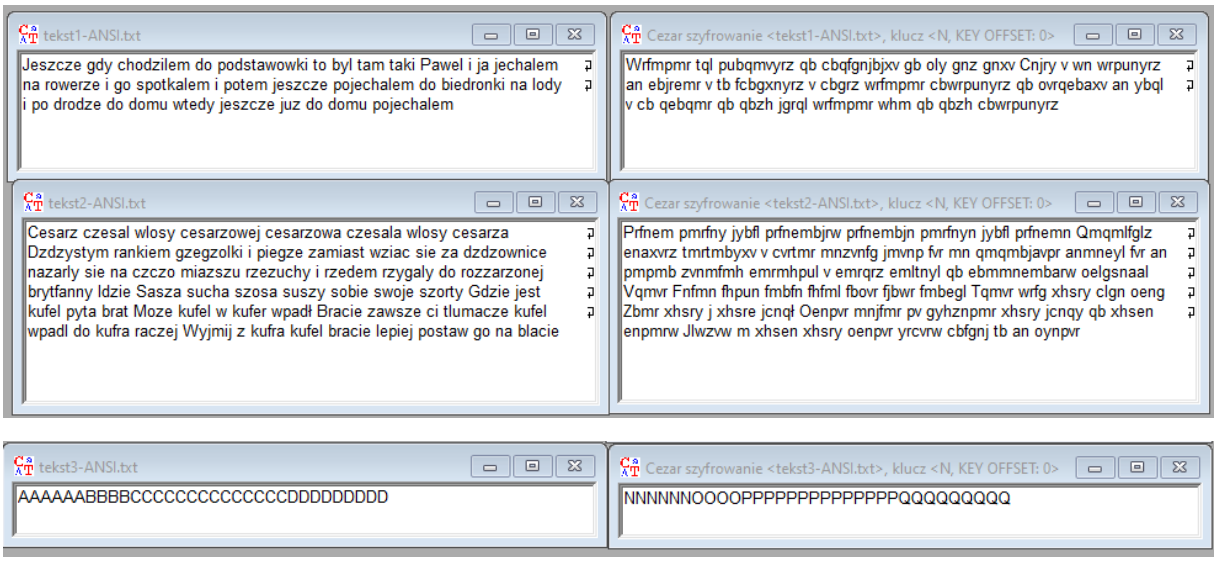
Algorytmy które wybrałem do analizy:

1) Cezar
2) Vigenere
3) Hill
4) Playfair
5) XOR
6) Permutacja / Transpozycja

Zadanie 1.1; 1.2:

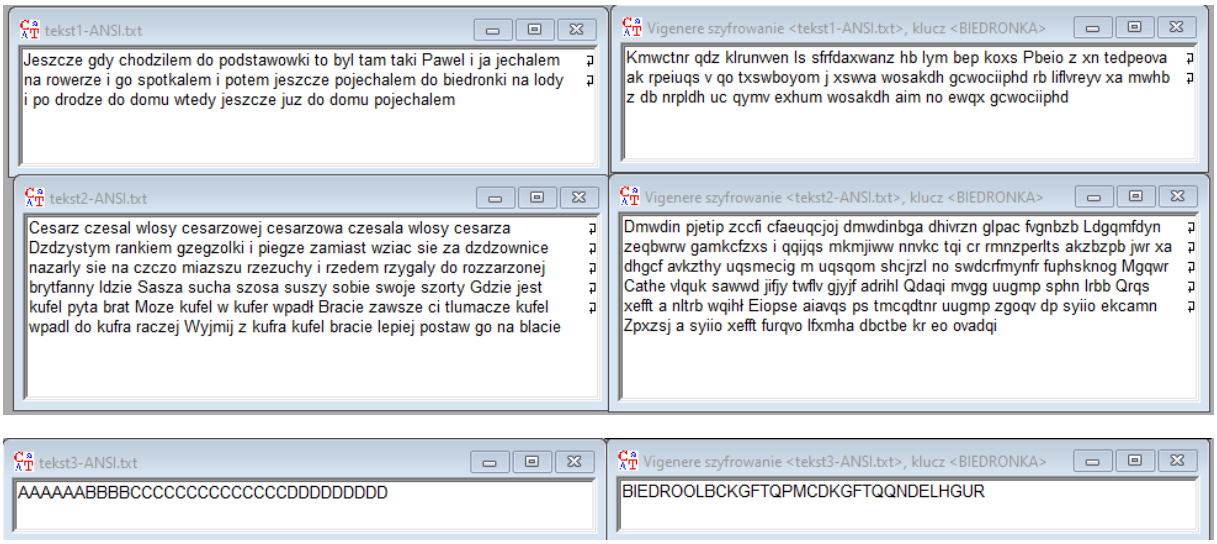
Cezar:

klucz = 13/„N”



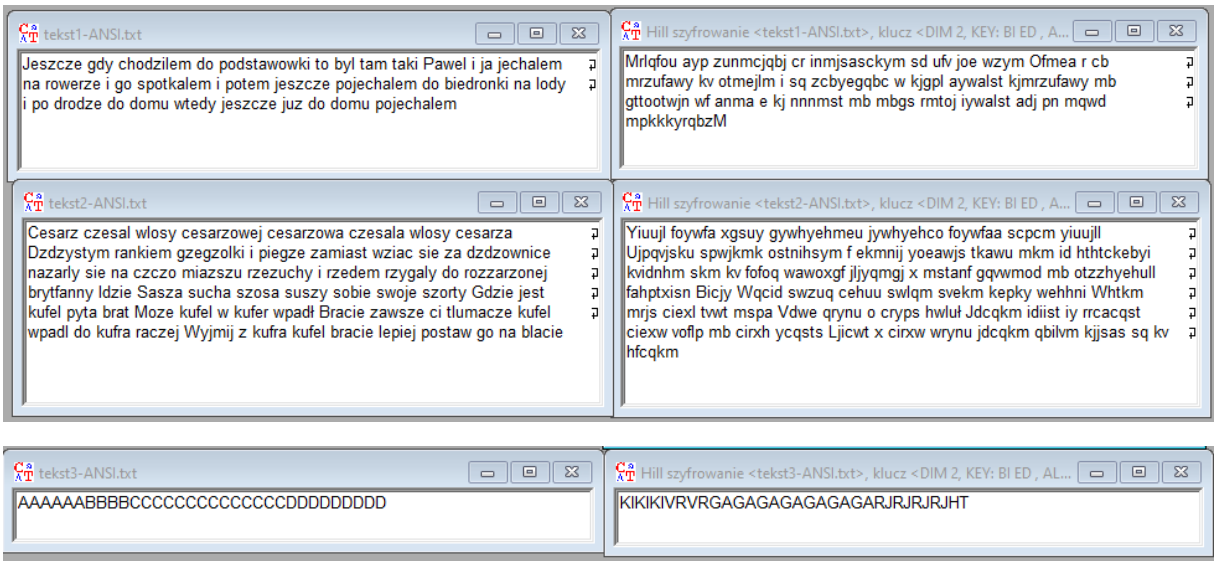
Vigenere:

klucz = „BIEDRONKA”



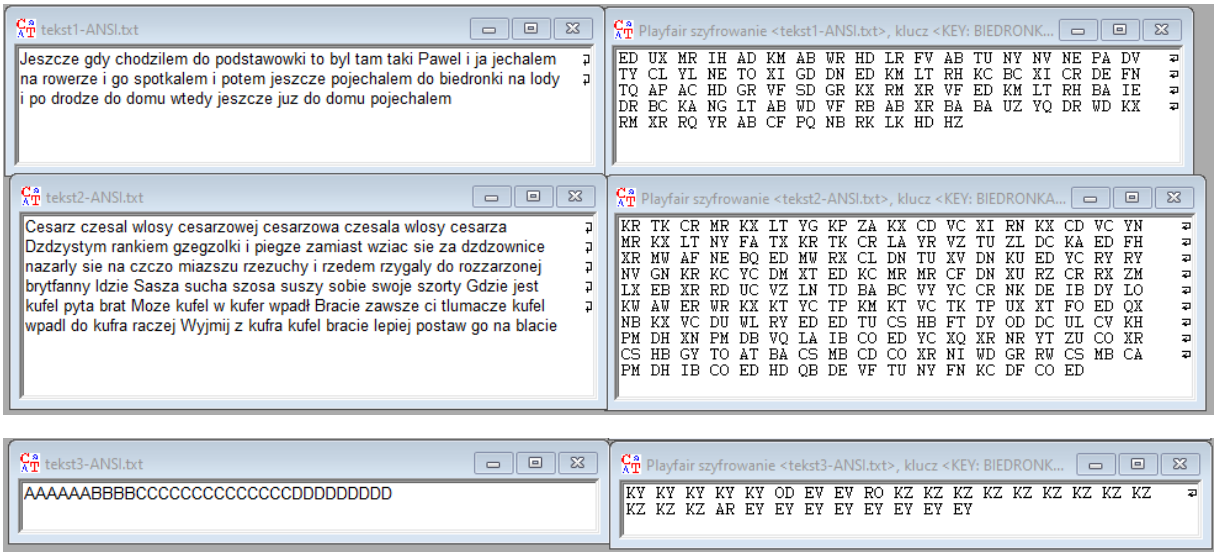
Hill:

klucz = „BI ED”, macierz 2x2



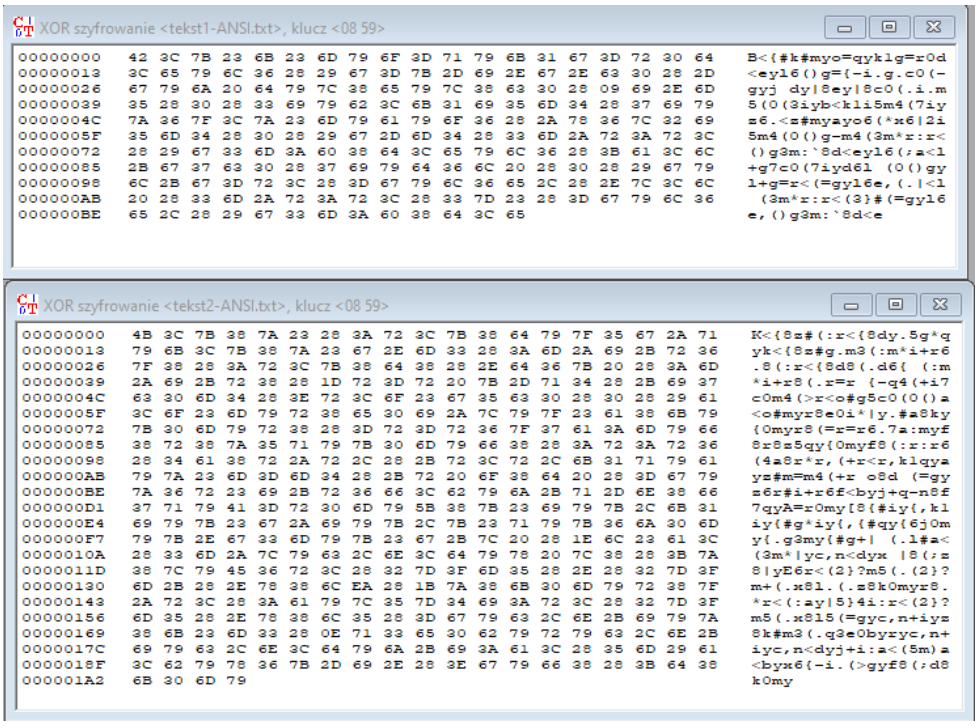
Playfair:

klucz = „BIEDRONKA”, macierz 5x5, po preformatowaniu



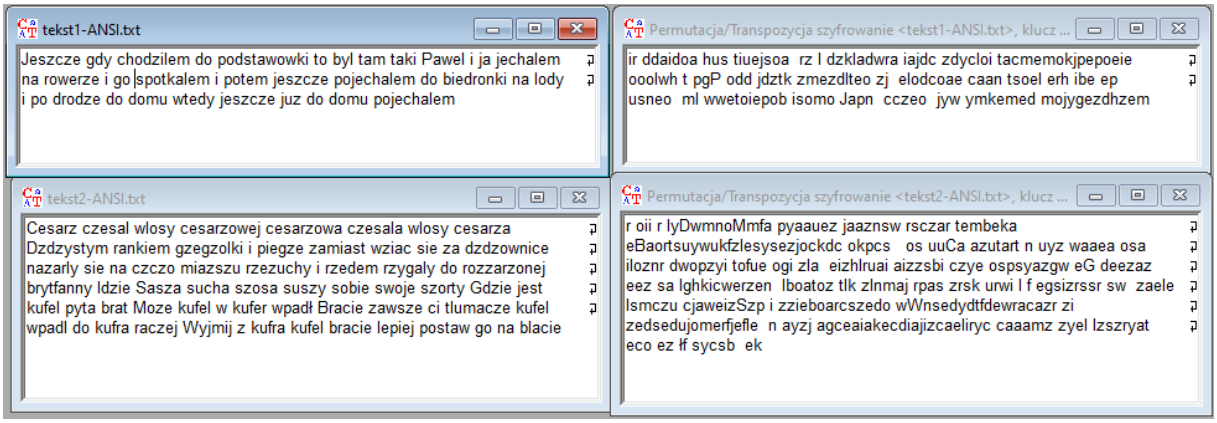
XOR:

klucz = 08 59



Permutacja / Transpozycja:

klucz = 1sza permutacja: „BIEDRONKA”, 2ga perm.: „ROWER” ((2, 5, 4, 3, 9, 8, 7, 6, 1); (3, 2, 5, 1, 4))



Wnioski:

1) **Algorytm Cezara** nie zmienił kształtu ani długości wejściowego tekstu. W każdym tekście można zauważyć powtarzające się litery na tych samych pozycjach, co w tekście jawnym. Zmiana wartości klucza jak i wielokrotne szyfrowanie nie daje w tym przypadku żadnej sensownej zmiany. Wynika to z prostego sposobu w jaki operuje ten algorytm. Szyfrowanie te jest bardzo proste do złamania, chociażby poprzez wykorzystanie analizy częstotliwości występowania liter w tekście zaszyfrowanym i porównanie ich z częstotliwością występowania w alfabecie.

2) **Algorytm Vigenera** nie zmienił kształtu ani długości wejściowego tekstu. Jednak w odróżnieniu od algorytmu Cezara nie jest tak łatwo znaleźć podobieństwa pomiędzy tekstem jawnym, a zaszyfrowanym ponieważ dwa takie same słowa mogą wyglądać inaczej w innym fragmencie tekstu. Dużą wadą tego algorytmu jest to, że może ujawnić klucz, na którym tekst został zaszyfrowany (najlepiej widać to na tekście numer 3, gdzie pierwsze kilka liter układa się w klucz, ponieważ dla litery ‘A’ algorytm dodaje 0 do kolejnych wartości klucza co skutkuje powtarzaniem klucza w tekście wyjściowym). Dla dłuższych kluczy (np. o długości tekstu jawnego) algorytm wydaje się być trudniejszy do złamania, jeśli klucz ten jest losowy i bezokresowy. Wielokrotne szyfrowanie wydłuża jedynie długość klucza.

3) **Algorytm Hilla** nie zmienił kształtu ani długości wejściowego tekstu. Jego szyfrowanie opiera się na matematycznej operacji macierzowej, która utrudnia analizę struktury tekstu. Jest bardziej odporny na analizę częstotliwości niż algorytmy Cezara czy Vigenera, jednak wymaga odpowiedniego doboru i odwracalności klucza macierzy, co może być wadą przy niewłaściwej implementacji. Dla przypadku tekstu numer 3 szyfr nie działa najlepiej, ponieważ wciąż widać powtarzalne fragmenty tekstu. Powiększenie macierzy kluczy nie dało lepszego rezultatu.

4) **Algorytm Playfair** grupuje tekst w pary liter, więc długość tekstu jawnego może się zmienić, ponieważ w przypadku wystąpienia podwójnych liter w parze (np. „ee”) wstawiana jest litera zapasowa (np. „x”). Dla przypadku tekstu numer 3 szyfr nie działa najlepiej, ponieważ wciąż widać powtarzalne fragmenty tekstu. Zmiana wartości klucza czy rozmiaru macierzy nie daje w tym przypadku żadnej sensownej zmiany.

5) **Algorytm XOR** zmienia długość tekstu, zależnie od zastosowanego klucza, który może mieć dowolną długość. Algorytm nie zachowuje struktury tekstu jawnego i przypomina swoją dwójkową strukturą algorytm Playfair. Jego siła wynika głównie z losowości klucza, jeśli klucz jest użyty jednorazowo i ma odpowiednią długość (podobnie jak w szyfrze Vernama), algorytm XOR może być teoretycznie nie do złamania. Dla przypadku tekstu numer 3 szyfr nie działa najlepiej, ponieważ wciąż widać powtarzalne fragmenty tekstu. Zmiana wartości klucza nie daje w tym przypadku żadnej sensownej zmiany, oprócz zwiększenia się okresu powtarzania znaków.

6) **Algorytm permutacji/transpozycji** – Algorytm ten zmienia jedynie kolejność liter lub bloków liter w tekście, co oznacza, że długość tekstu pozostaje niezmieniona. Jednak zmienia kształt tekstu w sposób znaczący, ponieważ układ liter jest mocno zmieniany. Permutacje mogą sprawić, że odnalezienie tekstu jawnego jest trudniejsze niż w przypadku prostych algorytmów podstawienia, takich jak Cezar czy Vigener. Dla przypadku tekstu numer 3 szyfr działa dużo lepiej niż pozostałe algorytmy, ponieważ pozbywa się jednorodności tekstu i sprawia wrażenie losowego (chociaż wciąż nie używa całego alfabetu znaków). Zmiana wartości klucza nie daje w tym przypadku żadnej sensownej zmiany.

Pytanie 1.3:

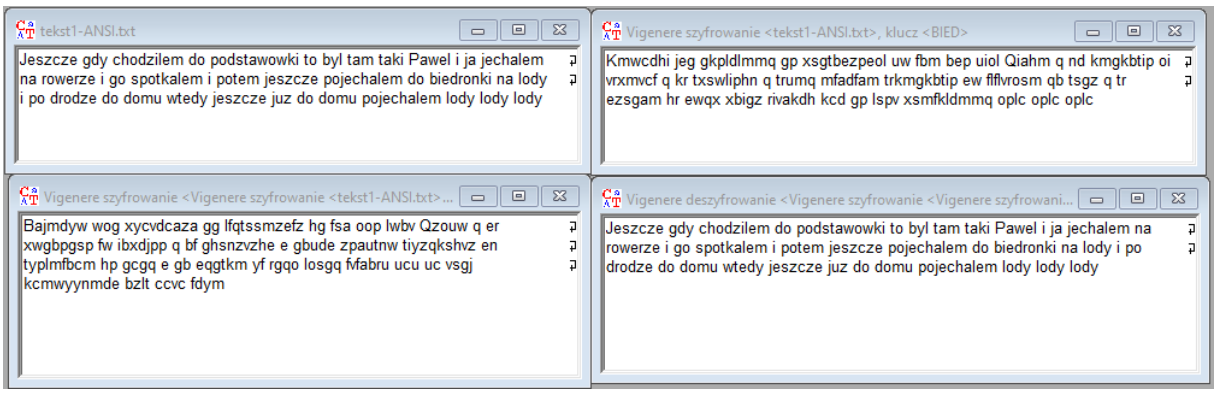
Szyfrowanie wielokrotne dla tych samych algorytmów co wcześniej.

Cezar:

Klucz1 = 13/„N”, Klucz2 = 4/”D”, klucz do deszyfrowania = 17/”Q”

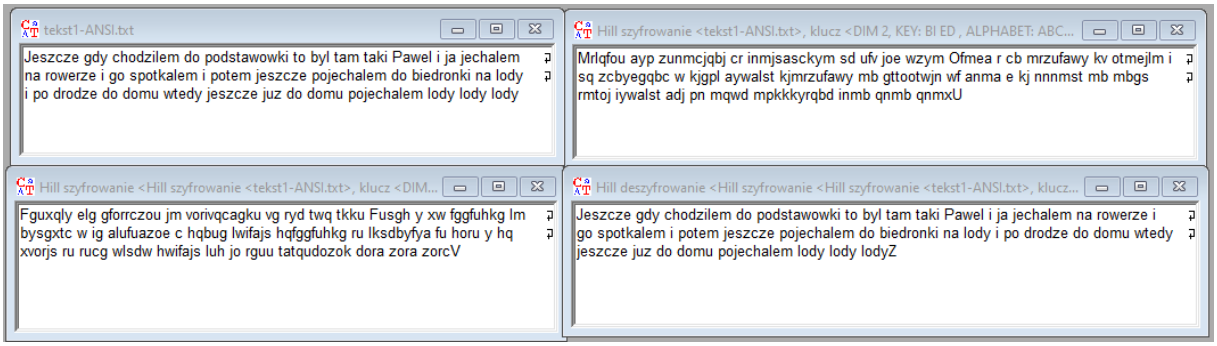
Vigenere:

Klucz1 = „BIED”, Klucz2 = „RONKA”, klucz do deszyfrowania = „SWRNBZSQLIVROSEUPVOD”

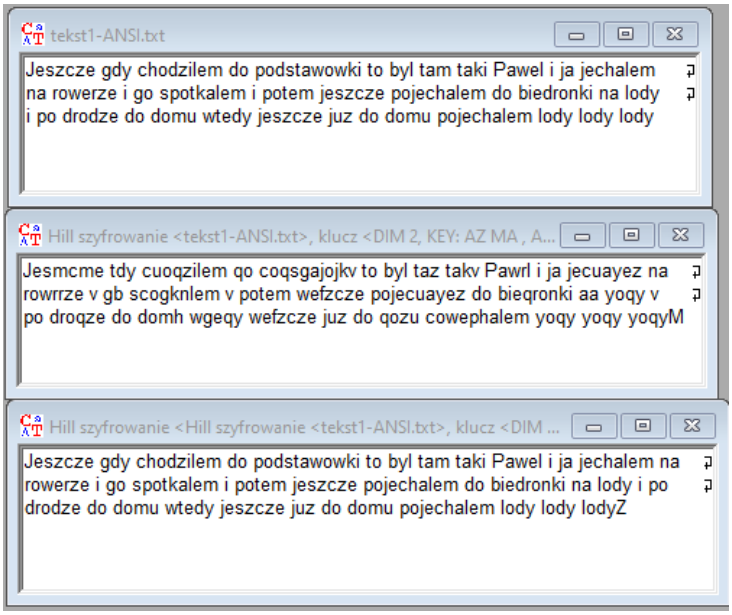


Hill:

Klucze 1 i 2 = „BI ED”, klucz deszyfrujący = „WD BI” macierz 2x2

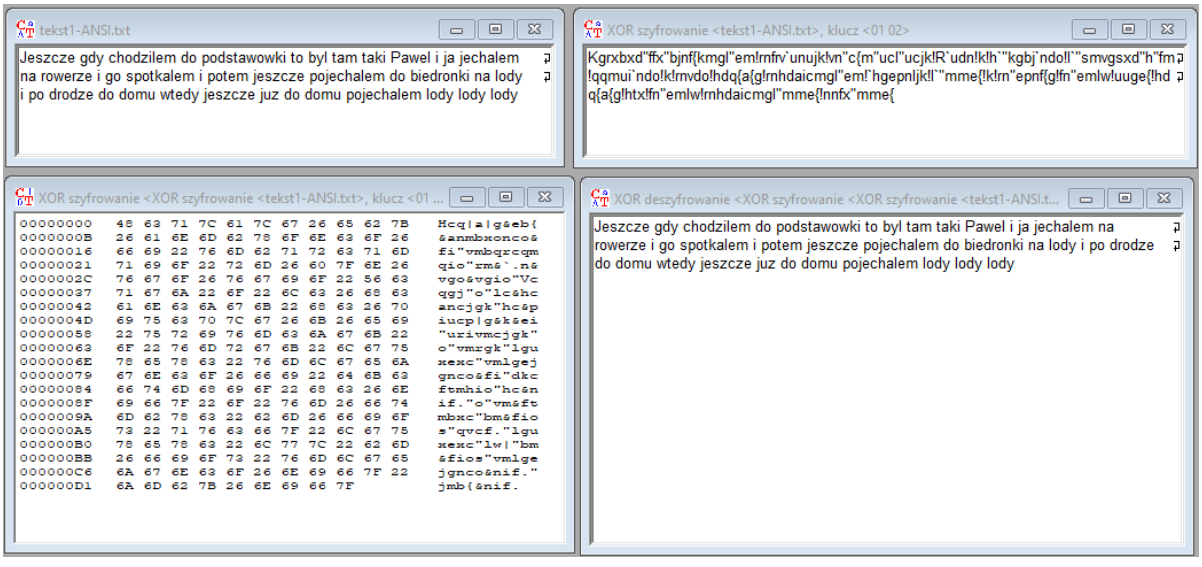


Klucze 1 i 2 = „AZ MA” macierz 2x2



XOR:

Klucz1 = „01 02”, klucz2 = „03 04”, klucz deszyfrujący: „02 06”



Wnioski:

Szyfrowanie wielokrotne **nie utrudnia zbytnio procesu rozszyfrowania tekstu**. W przypadku szyfrów takich jak szyfr Cezara (korzystający z jednego alfabetu z przesunięciem modulo 26) wielokrotne szyfrowanie będzie mieć takie same skutki co szyfrowanie jednokrotnie o inną długość klucza. W przypadku pozostałych algorytmów takich jak szyfr Vigenera (korzystających z wielu alfabetów) wielokrotne szyfrowanie może mieć niewielkie znaczenie, bo sprawia jedynie że komplikuje się klucz służący do odszyfrowania wiadomości.

W przypadku szyfru Hilla wielokrotne szyfrowanie czasem utrudnia proces rozszyfrowania, bo może się znacznie skomplikować klucz deszyfrujący, lecz czasami szyfrowanie wielokrotne powoduje całkowite rozszyfrowanie tekstu (szczególnie dobrze widać to na przykładzie drugim, gdzie po podwójnym zaszyfrowaniu przy pomocy klucza matrycy 2x2 (AZ MA) tekst wraca do oryginalnego stanu i nie jest wcale zaszyfrowany). Wielokrotne szyfrowanie z użyciem XOR z tym samym kluczem nie ma sensu, ponieważ podwójne szyfrowanie XOR jest równoważne z odszyfrowaniem tekstu.

Podsumowując efekty wielokrotnego szyfrowania mają niewielki (Vigener, Hill) lub zerowy (Cezar, XOR) wpływ na trudność procesu rozszyfrowania tekstu w porównaniu do tekstu zaszyfrowanego jednokrotnie.

Pytanie 1.4:

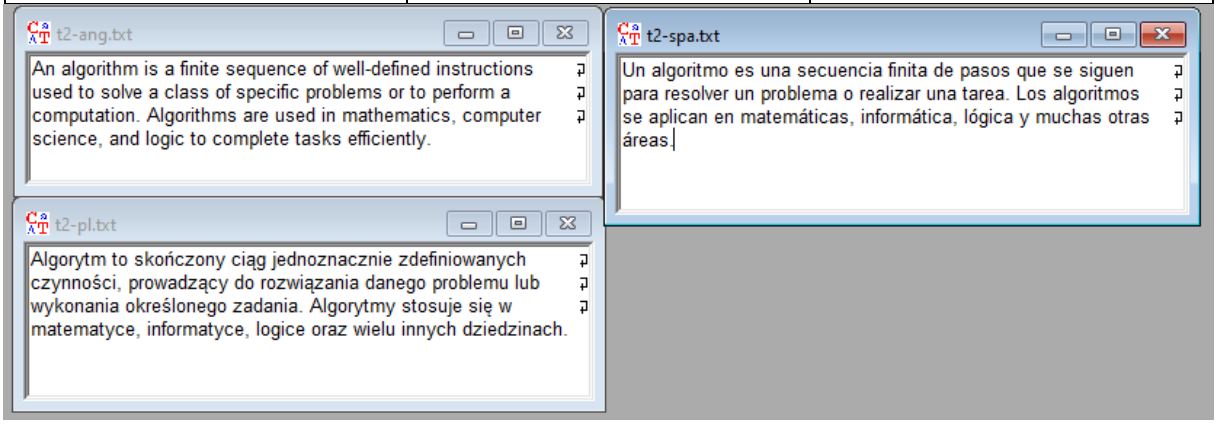
Z przetestowanych wcześniej algorytmów można uznać **Vigenere** oraz **Hill** za silniejsze od reszty, zwłaszcza przy odpowiednich ustawieniach kluczy. Vigenere z długim, losowym kluczem podobnie jak Hill z dobrze dobraną macierzą jest trudniejszy do złamania niż Cezar, który łatwo złamać przeprowadzając prostą analizę frekwencji występowania znaków. **XOR** mimo swojej prostoty jest bezpieczny tylko wtedy, gdy klucz jest równie długi jak tekst, w innym przypadku jest podatny na ataki.

Zad 2. Analiza własności dostępnych algorytmów.

Zadanie 2.1:

Przetestowane na podstawie fragmentu z artykułu "Algorytm" na Wikipedii.

Entropia tekstów jawnych dla języków: polski, angielski, hiszpański		
Polski	**Angielski**	**Hiszpański**
4,19	**4,06**	**3,93**

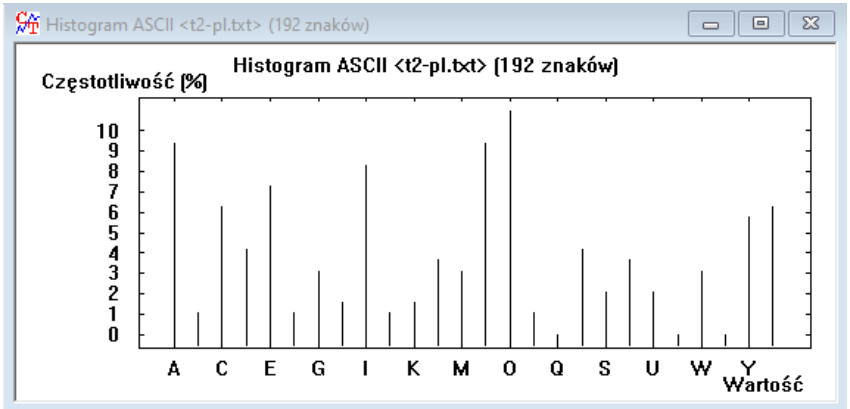


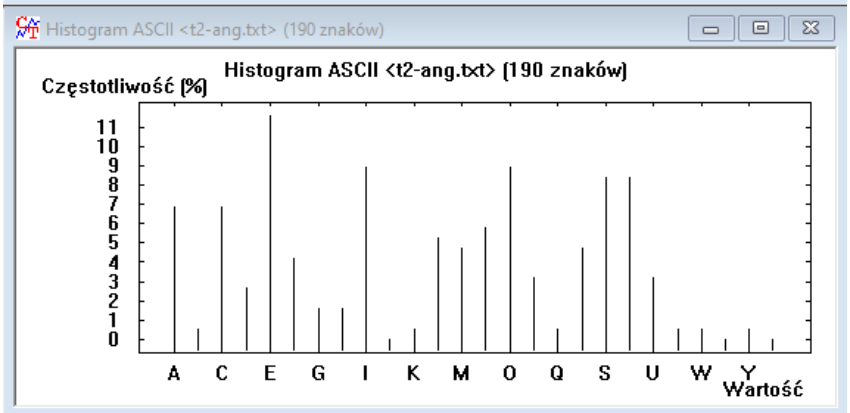
Zadanie 2.2:

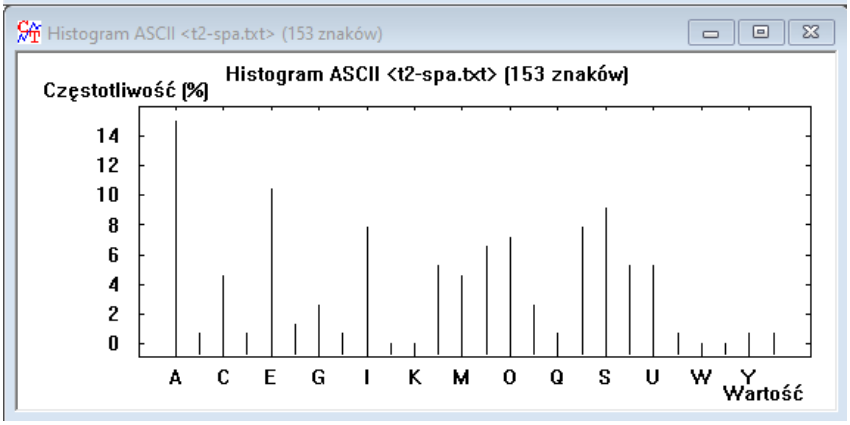
Entropia tekstu jawnego i tekstu zaszyfrowanego (dla tekstu polskiego)	
Cezar	4,19/4,7
Adfgvx	2,57/4,7
Homofony	6,86/8,0 (znormalizowane: 4,03)
Permutacyjny	4,19/4,7
Vigenere (klucz: AE)	4,44/4,7
Vigenere (klucz: BIEDRONKA)	4,62/4,7
Vigenere (klucz: KROLKAROLKUPILKROLOWEJ)	4,53/4,7
Hilla (klucz: BI ED)	4,56/4,7
Hilla (klucz: JAC ABA CAK)	4,62/4,7
Hilla (klucz: SVBQH HLIMN ASMCF XVHML FRYAA)	4,57/4,7

Zadanie 2.3:

Histogramy ASCII kolejno dla języków: polski, angielski, hiszpański

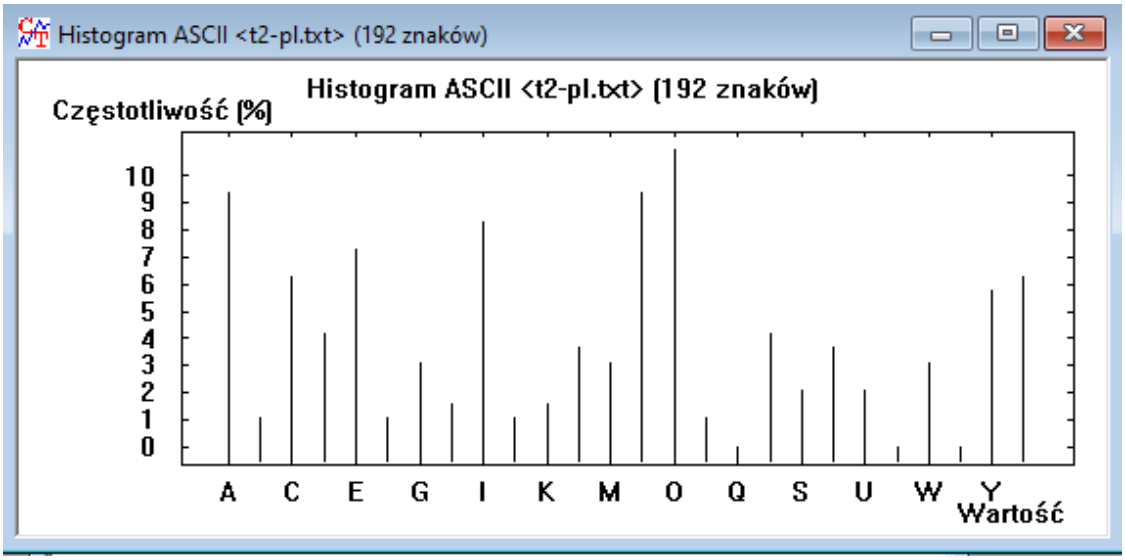




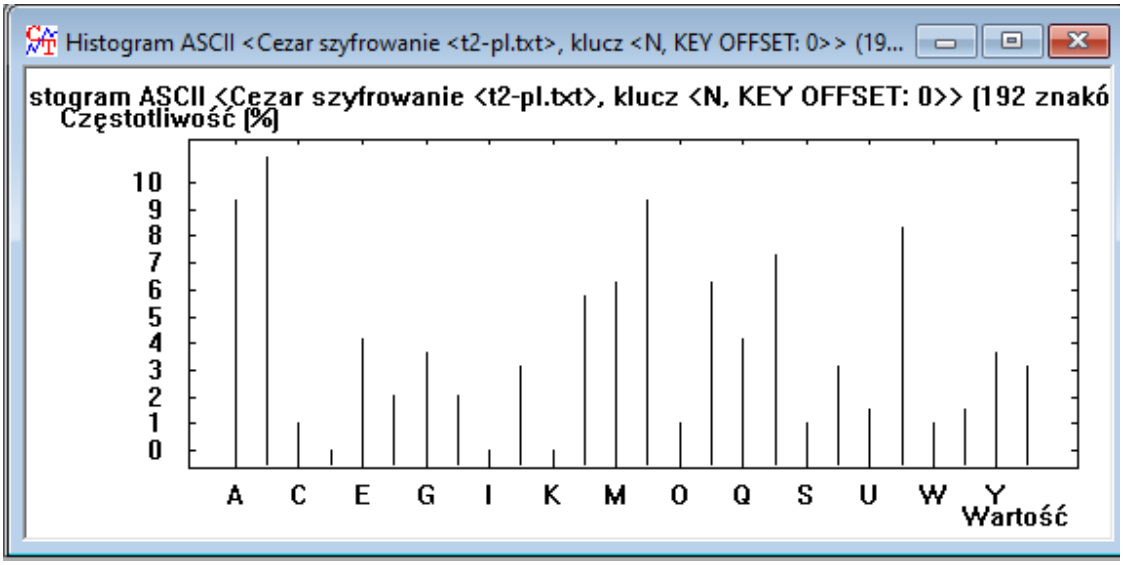


Zadanie 2.4:

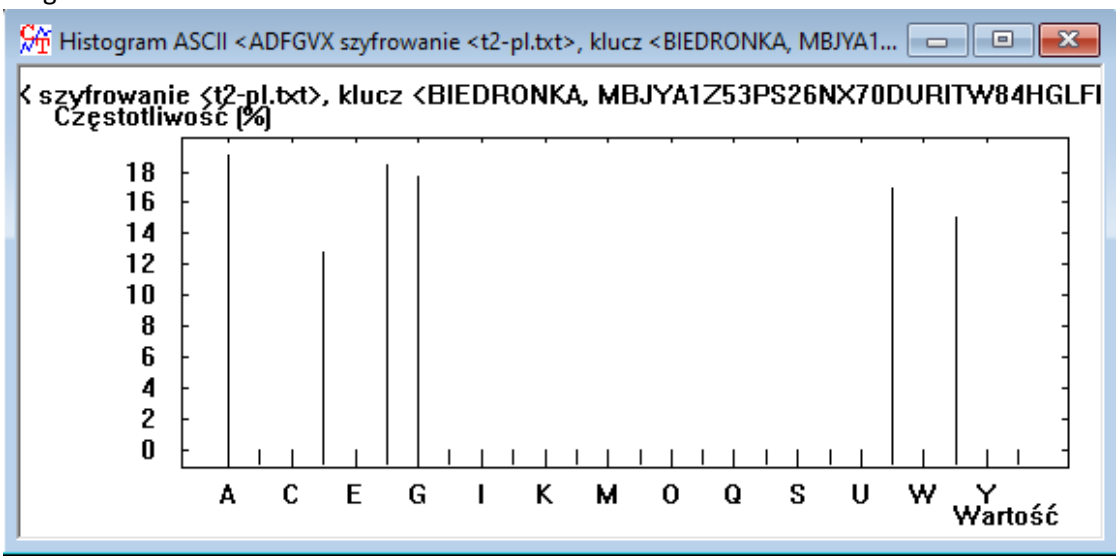
1) Tekst jawny:



2) Cezar

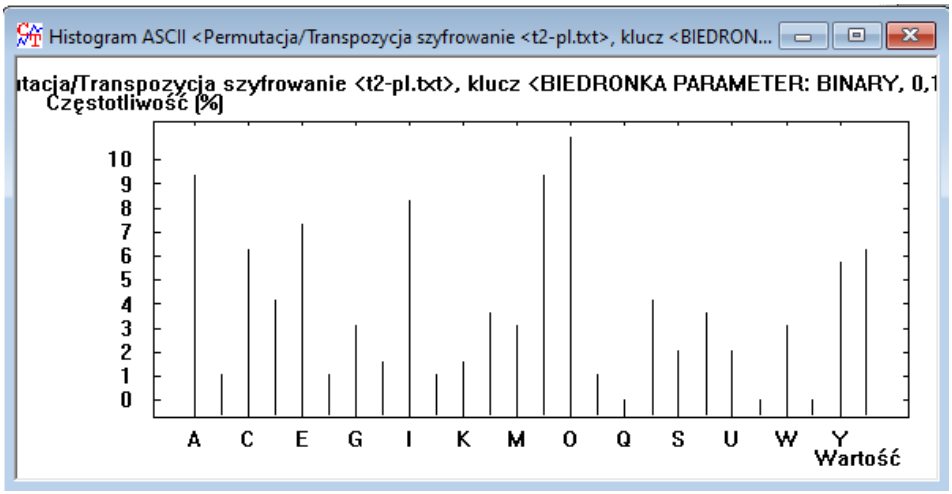


3) Adfgvx:

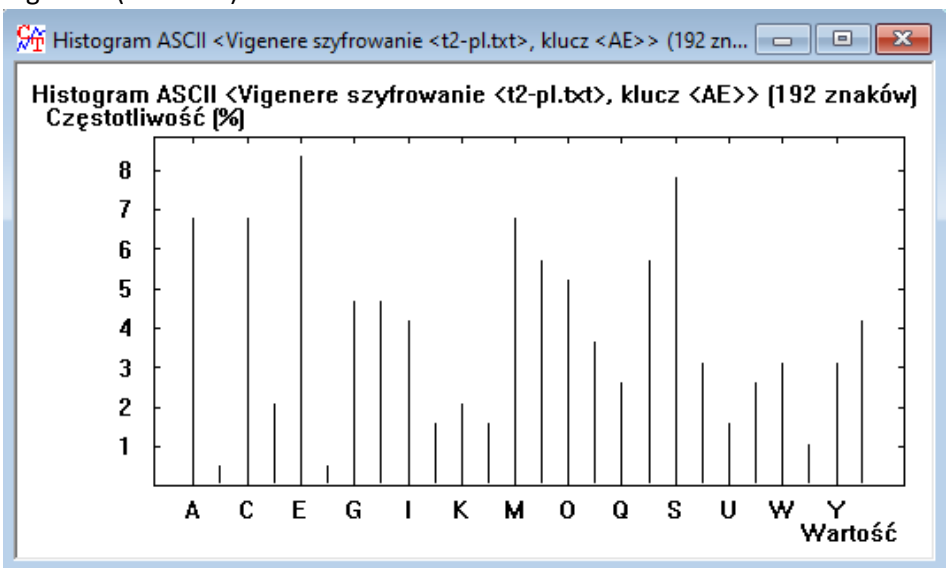


4) Homofony: Histogramu dla homofonów nie udało się wygenerować z uwagi na błąd w programie Cryptool.

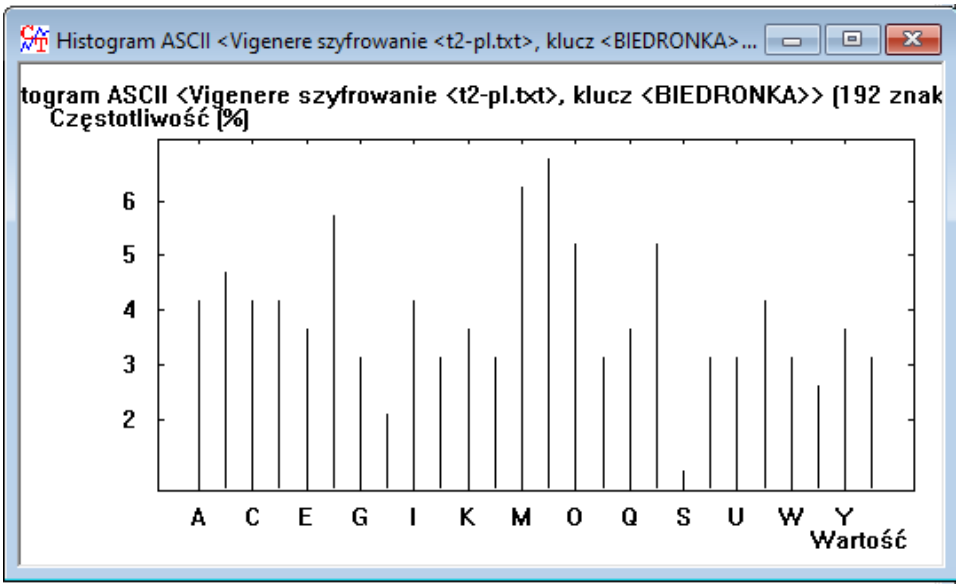
5) Permutacyjny



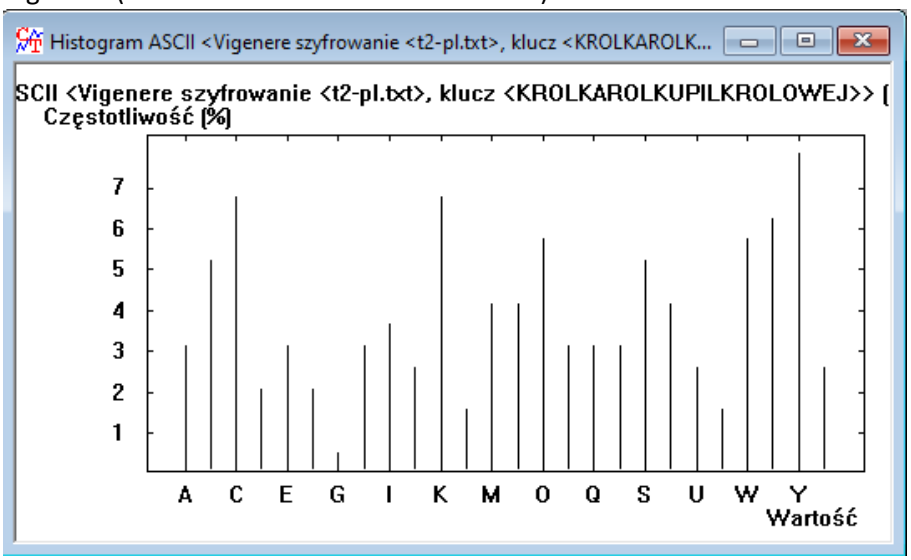
6) Vigenere (klucz: AE)



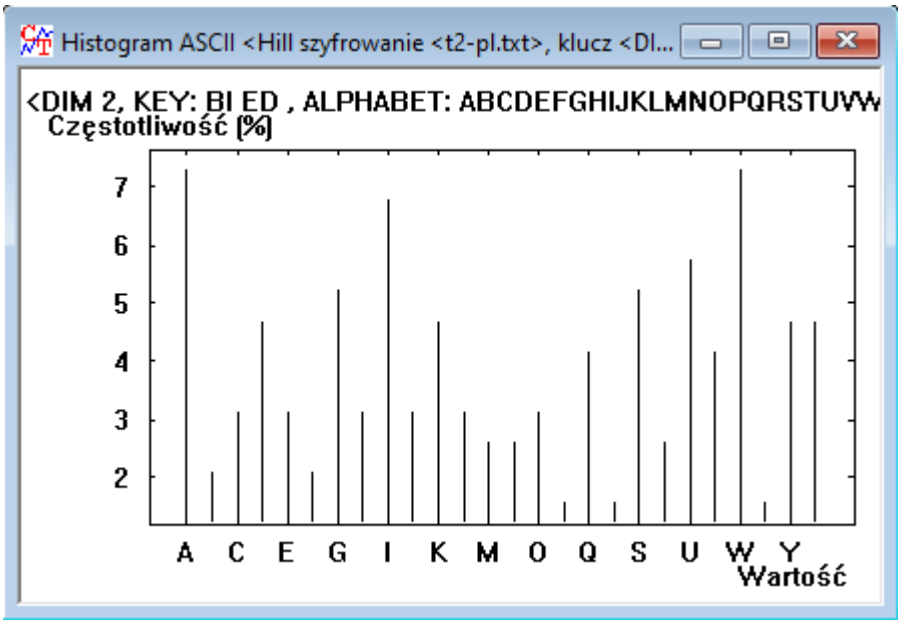
7) Vigenere (klucz: BIEDRONKA)



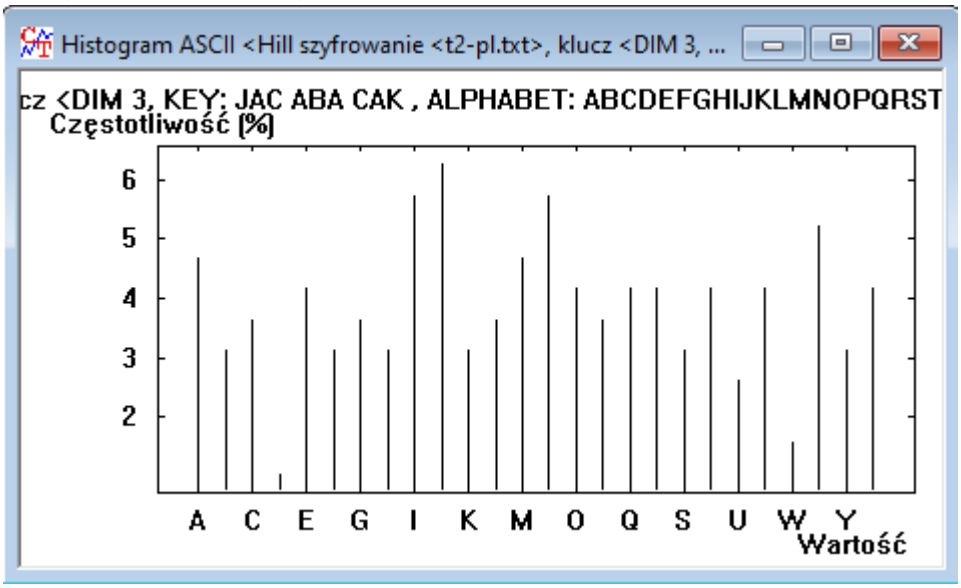
8) Vigenere (klucz: KROLKAROLKUPILKROLOWEJ)



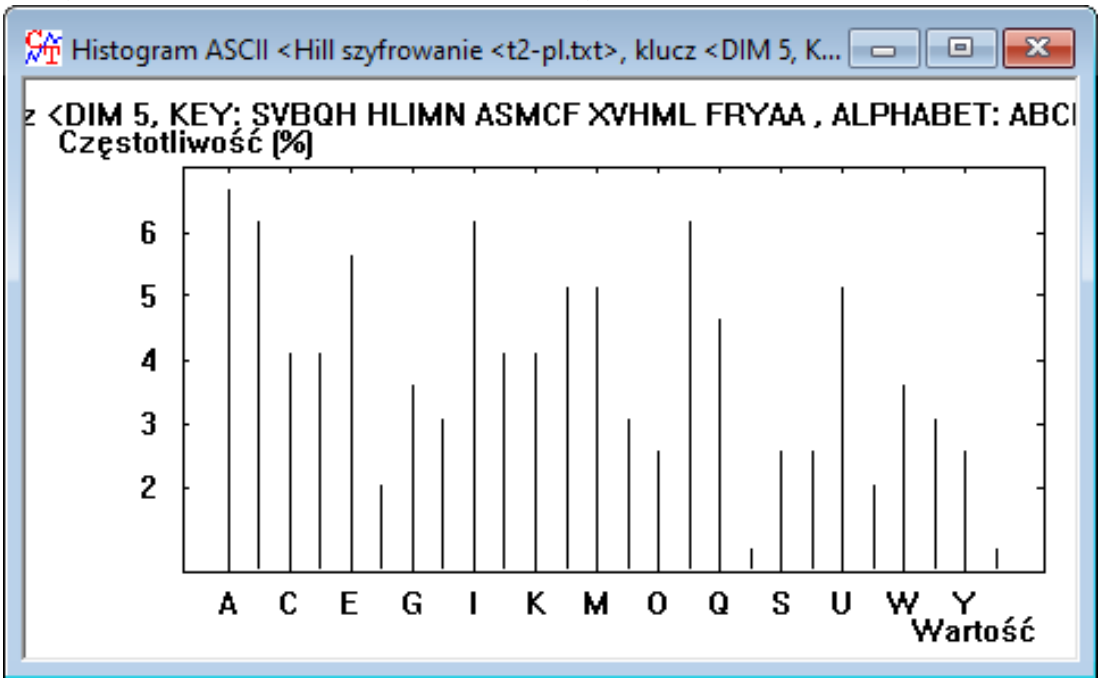
9) Hilla (klucz: BI ED)



10) Hilla (klucz: JAC ABA CAK)



11) Hilla (klucz: SVBQH HLIMN ASMCF XVHML FRYAA)



Zadanie 2.5:

Najczęstsze Digramy, Trigramy i N-gramy (dla tekstu numer 2 przetłumaczonego na inne języki) kolejno w języku:

a) Polskim:

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	A	12.0352	123
2	E	9.2955	95
3	R	8.7084	89
4	Z	6.4579	66
5	O	5.9687	61

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	AR	3.9735	30
2	RZ	3.7086	28
3	IE	2.9139	22
4	TA	2.9139	22
5	ZY	2.2517	17

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	RZY	2.0599	11
2	ARZ	1.6854	9
3	RZE	1.6854	9
4	PCH	1.4981	8
5	EWO	1.3109	7

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	EWOL	1.9499	7
2	LWER	1.9499	7
3	OLWE	1.9499	7
4	REWO	1.9499	7
5	WOLW	1.9499	7
6	EROW	1.6713	6

b) Angielskim:

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	E	13.8153	172
2	T	8.9960	112
3	A	7.9518	99
4	R	7.9518	99
5	O	7.0683	88

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	TH	4.8626	46
2	HE	4.2283	40
3	ER	3.0655	29
4	IN	3.0655	29
5	RE	2.5370	24

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	THE	5.1929	35
2	ING	1.4837	10
3	HER	1.3353	9
4	FLE	1.0386	7
5	MBE	1.0386	7

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	THER	1.5766	7
2	FLEA	1.3514	6
3	MBER	1.1261	5
4	EASE	0.9009	4
5	EVOL	0.9009	4

c) Hiszpańskim:

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	A	15.8527	211
2	E	12.6221	168
3	R	8.7153	116
4	L	8.3396	111
5	O	8.3396	111

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	LA	3.7111	37
2	RA	3.3099	33
3	AS	3.2096	32
4	ER	3.2096	32
5	DE	2.9087	29

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	ASA	1.3043	9
2	QUE	1.3043	9
3	ERA	1.1594	8
4	RAS	1.1594	8
5	ERR	1.0145	7

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	RASA	1.4737	7
2	ASER	1.2632	6
3	ERRA	1.2632	6
4	PULG	1.2632	6
5	SERR	1.2632	6

Zadanie 2.6:

Najczęstsze Digramy, Trigramy i N-gramy dla tekstu numer 2 zaszyfrowanego przez:

a) Nic

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	A	12.0352	123
2	E	9.2955	95
3	R	8.7084	89
4	Z	6.4579	66
5	O	5.9687	61

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	AR	3.9735	30
2	RZ	3.7086	28
3	IE	2.9139	22
4	TA	2.9139	22
5	ZY	2.2517	17

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	RZY	2.0599	11
2	ARZ	1.6854	9
3	RZE	1.6854	9
4	PCH	1.4981	8
5	EWO	1.3109	7

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	EWOL	1.9499	7
2	LWER	1.9499	7
3	OLWE	1.9499	7
4	REWO	1.9499	7
5	WOLW	1.9499	7
6	EROW	1.6713	6

b) Cezara

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	N	12.0352	123
2	R	9.2955	95
3	E	8.7084	89
4	M	6.4579	66
5	B	5.9687	61

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	NE	3.9735	30
2	EM	3.7086	28
3	GN	2.9139	22
4	VR	2.9139	22
5	ML	2.2517	17

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	EML	2.0599	11
2	EMR	1.6854	9
3	NEM	1.6854	9
4	CPU	1.4981	8
5	BYJ	1.3109	7

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	BYJR	1.9499	7
2	ERJB	1.9499	7
3	JBYJ	1.9499	7
4	RJBY	1.9499	7
5	YJRE	1.9499	7

c) Adfgvx

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	A	28.0333	573
2	F	21.4286	438
3	D	16.9276	346
4	V	15.1174	309
5	G	12.3288	252
6	X	6.1644	126

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	AA	8.3211	170
2	AF	5.7269	117
3	FA	5.5311	113
4	VA	4.7479	97
5	AD	4.6990	96
6	DA	4.6011	94

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	AAA	2.9873	61
2	AAF	1.6650	34
3	FAA	1.5671	32
4	AFA	1.5181	31
5	VAA	1.5181	31
6	FAF	1.4691	30

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	AAAA	0.7349	15
2	FAAA	0.6859	14
3	AAAF	0.6369	13
4	AAAG	0.6369	13
5	VAAA	0.6369	13
6	AAAV	0.4900	10

d) Homofony – nie wygenerowało

e) Permutacyjny

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	A	12.0352	123
2	E	9.2955	95
3	R	8.7084	89
4	Z	6.4579	66
5	O	5.9687	61

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	RW	1.8843	14
2	AT	1.7497	13
3	AC	1.3459	10
4	EA	1.3459	10
5	WR	1.3459	10

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	RWO	0.5556	3
2	RWR	0.5556	3
3	WRW	0.5556	3
4	ATE	0.3704	2
5	ATK	0.3704	2

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	WRWR	0.5000	2
2	AAAJ	0.2500	1
3	AAJR	0.2500	1
4	AASU	0.2500	1
5	ABET	0.2500	1

f) Vigenere (klucz: AE)

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	E	11.3503	116
2	A	8.0235	82
3	I	6.8493	70
4	W	5.8708	60
5	R	5.6751	58

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	AV	2.5166	19
2	ME	2.5166	19
3	ER	1.9868	15
4	VZ	1.9868	15
5	TE	1.8543	14

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	RDY	1.1236	6
2	VZI	1.1236	6
3	AEV	0.9363	5
4	ERD	0.9363	5
5	IWS	0.9363	5

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	IWSL	1.3928	5
2	LAEV	1.3928	5
3	RIWS	1.3928	5
4	SLAE	1.3928	5
5	WSLA	1.3928	5

g) Vigenere (klucz: BIEDRONKA)

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	I	6.3601	65
2	B	5.2838	54
3	S	5.1859	53
4	E	4.7945	49
5	R	4.7945	49

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	BZ	1.0596	8
2	NB	1.0596	8
3	GK	0.9272	7
4	TI	0.9272	7
5	WR	0.9272	7

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	GKR	0.5618	3
2	INL	0.5618	3
3	SHI	0.5618	3
4	STV	0.5618	3
5	UQM	0.5618	3

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	BEXW	0.5571	2
2	BGAO	0.5571	2
3	EFSI	0.5571	2
4	EXWP	0.5571	2
5	FBGA	0.5571	2

h) Vigenere (klucz: KROLKAROLKUPILKROLOWEJ)

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	K	6.3601	65
2	P	5.3816	55
3	O	5.2838	54
4	N	4.7945	49
5	I	4.5010	46

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	IN	1.0596	8
2	EK	0.7947	6
3	FP	0.7947	6
4	KI	0.7947	6
5	NJ	0.7947	6

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	INJ	0.7491	4
2	FZH	0.5618	3
3	GFZ	0.5618	3
4	GQS	0.5618	3
5	INP	0.5618	3

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	GFZH	0.8357	3
2	FZHO	0.5571	2
3	GLBZ	0.5571	2
4	MVGL	0.5571	2
5	NGFZ	0.5571	2

i) Hilla (klucz: BI ED)

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	Y	7.9256	81
2	W	6.6536	68
3	I	5.6751	58
4	Q	5.6751	58
5	S	5.6751	58

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	HY	2.5166	19
2	JL	1.7219	13
3	KM	1.7219	13
4	WZ	1.5894	12
5	IQ	1.4570	11

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	CFY	0.9363	5
2	FYS	0.9363	5
3	SWV	0.9363	5
4	VPS	0.9363	5
5	WVP	0.9363	5

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	CFYS	1.3928	5
2	FYSW	1.3928	5
3	SWVP	1.3928	5
4	WVPS	1.3928	5
5	YSWV	1.3928	5

j) Hilla (klucz: JAC ABA CAK)

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	U	5.9629	61
2	J	4.8876	50
3	L	4.8876	50
4	N	4.7898	49
5	T	4.6921	48

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	BK	1.0596	8
2	WE	0.9272	7
3	NY	0.7947	6
4	UB	0.7947	6
5	UN	0.7947	6

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	MWE	0.9363	5
2	UNY	0.9363	5
3	BKU	0.7491	4
4	UQQ	0.7491	4
5	ZGZ	0.7491	4

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	HCYL	0.8357	3
2	MHCY	0.8357	3
3	UMHC	0.8357	3
4	BKJD	0.5571	2
5	CYLT	0.5571	2

k) Hilla (klucz: SVBQH HLIMN ASMCF XVHML FRYAA)

Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	Q	5.7561	59
2	Z	5.2683	54
3	Y	4.9756	51
4	M	4.8780	50
5	H	4.5854	47

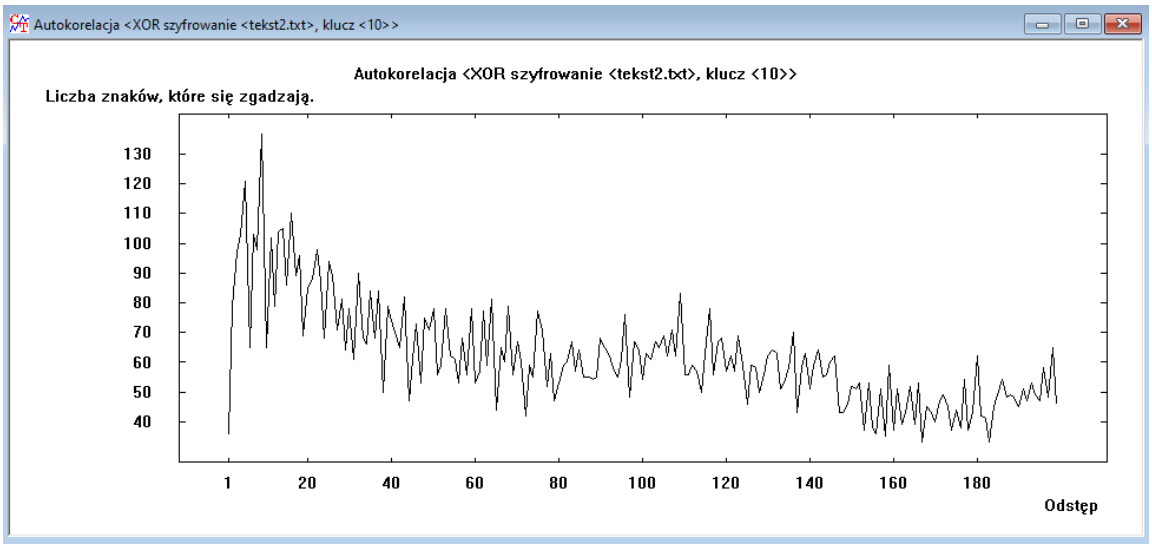
Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	CU	0.7926	6
2	GM	0.7926	6
3	ZT	0.7926	6
4	HS	0.6605	5
5	HZ	0.6605	5

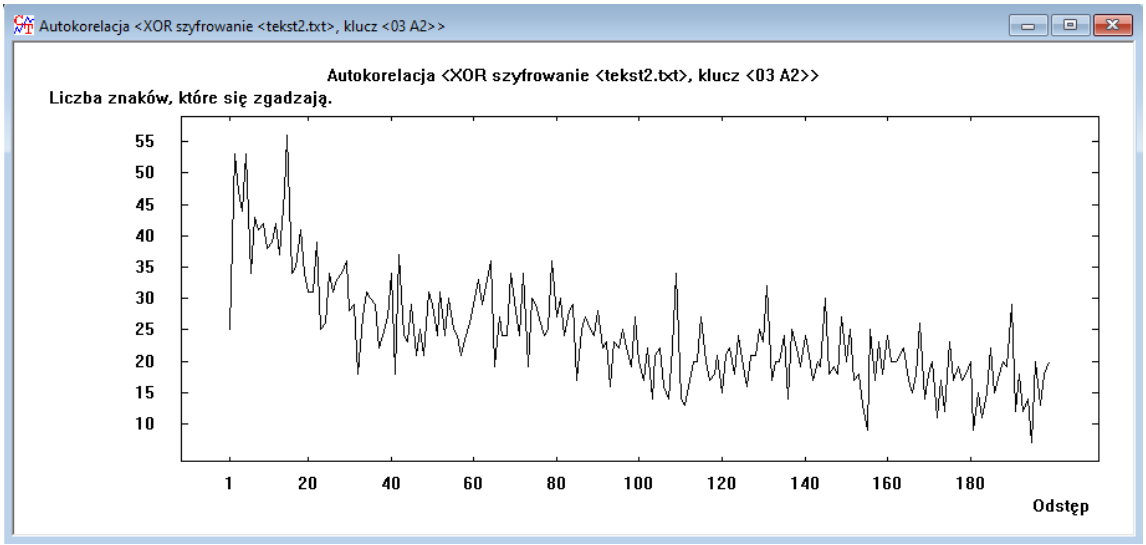
Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	CHZ	0.7477	4
2	HZT	0.7477	4
3	ZCH	0.7477	4
4	BZN	0.5607	3
5	HSG	0.5607	3

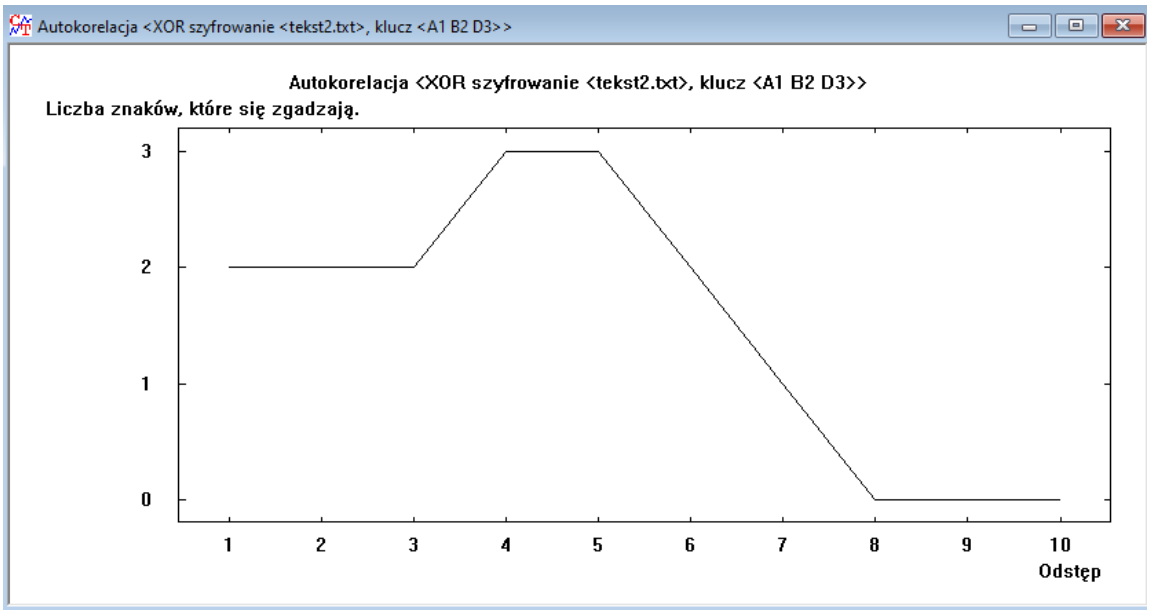
Nr.	Sekwencja zn...	Częstotliwość ...	Częstotliwość
1	CHZT	1.1142	4
2	ZCHZ	1.1142	4
3	BZNQ	0.8357	3
4	HSGM	0.8357	3
5	ZNQD	0.8357	3

Zadanie 2.7:

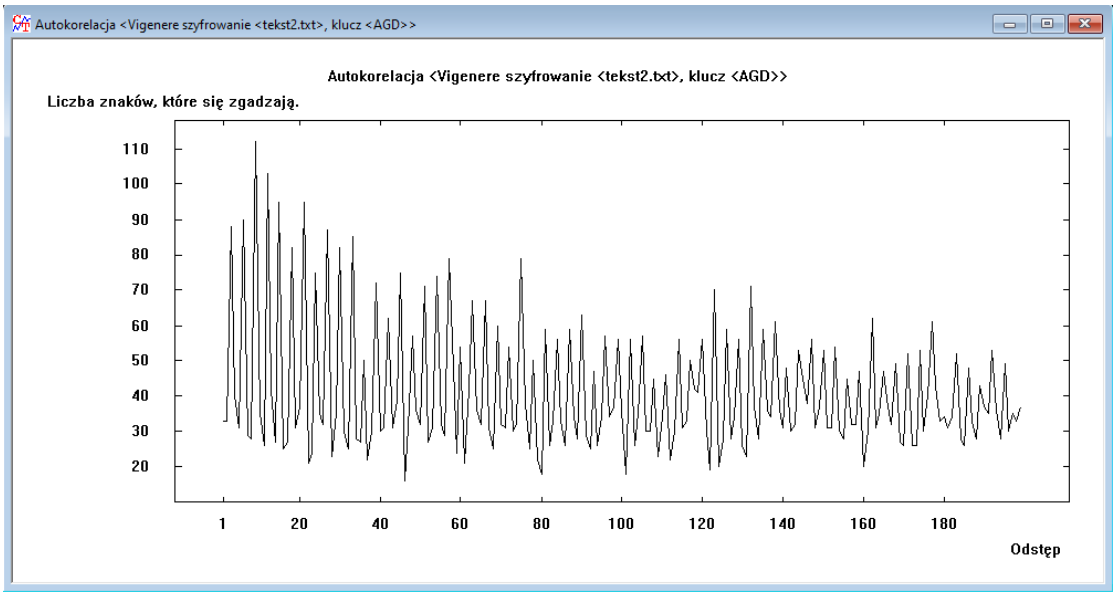
Autokorelacja dla algorytmu szyfrującego XOR oraz rosnącą długością klucza:

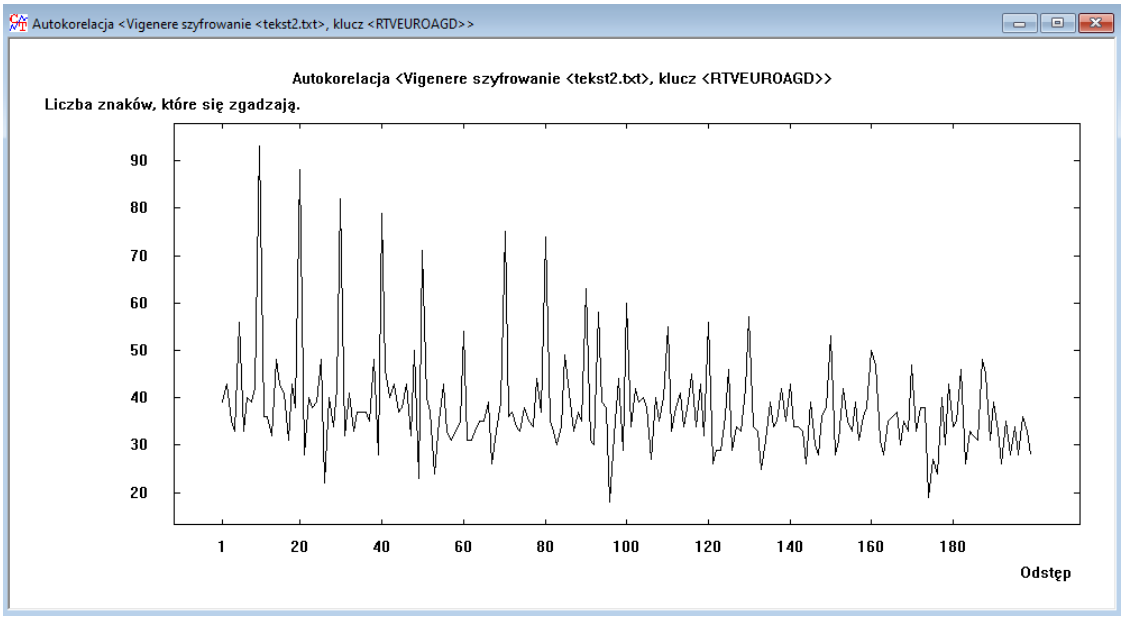


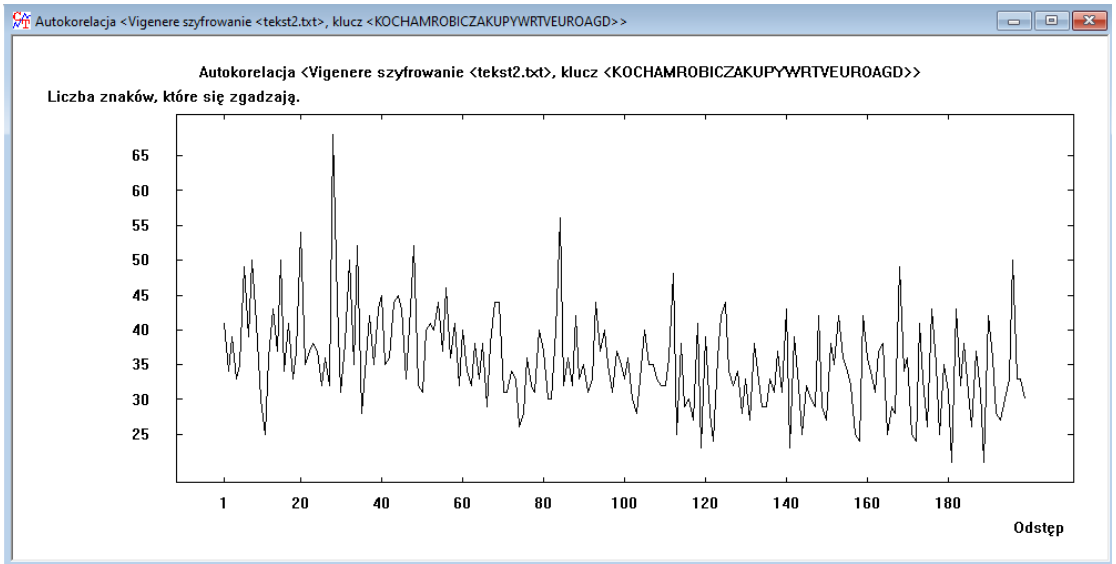




Autokorelacja dla algorytmu szyfrującego Vigenera oraz rosnącą długością klucza:







Pytanie 2.8:

1) Największą entropię spośród analizowanych języków posiada język polski (4,19), natomiast najmniejszą język hiszpański (3,93)
2) Dla szyfru Cezara i permutacyjnego entropia nie ulega zmianie, ponieważ nie tracimy żadnej informacji, jedynie przesuwamy każdy znak o ten sam klucz w przypadku cezara lub zamieniamy znaki miejscami w przypadku permutacji. Dla szyfru ADFGVX entropia spada znacznie, ponieważ używamy tylko 6 znaków. Dla homofonów entropia rośnie, bo pojawiają się nowe znaki. Dla Vigenere i Hilla entropia wzrasta, wraz z wzrostem długości klucza / wzrostem rozmiaru macierzy.
3) We wszystkich tekstach najczęściej występowały samogłoski, z drobnymi różnicami w częstotliwości np. w języku angielskim najczęstszą samogłoską było „E" natomiast w hiszpańskim oraz polskim było to „A". Z kolei w języku polskim jako jedynym tak często padały litery „Y" oraz „Z"
4) Dla szyfru cezara nie zmienił kształtu, tylko przesunął się na osi X. Dla szyfru adgfvx widać tylko 6 liter, które się powtarzają. Permutacyjny jest bez zmian z tekstem jawnym. Vigener oraz Hill posiadają coraz bardziej płaskie histogramy wraz ze wzrostem rozmiaru klucza (długości klucza lub rozmiaru matrycy)
5) Można z łatwością rozpoznać pewne konstrukcje językowe dostępne tylko w konkretnych językach, które dużo rzadziej pojawiają się w innych językach (np. „PCH" „RZY" które są konstrukcjami charakteryzującymi język polski)
6) Dla cezara są te same konstrukcje co w tekście jawnym, ale przesunięte. Dla permutacyjnego konstrukcje jednoznakowe zostają takie same, ale wszystkie dłuższe zostają zniszczone i zostaje na ich miejscu „szum".
7) Dla XORa liczba pasujących znaków maleje wraz z wzrostem długości klucza. Okresowości dla tego szyfru są raczej niewidoczne. Vigener dla krótkich kluczy ujawnia swoją okresowość, wraz ze zwiększaniem długości klucza problem ten zanika i staje się mniej dostrzegalny.

Pytanie 2.9:

1) Pomiar entropi – wysoki jej pomiar może oznaczać algorytm Hilla z macierzą o dużych rozmiarach lub Vigenera o długim kluczu, mała entropia natomiast może oznaczać adfgvx, natomiast entropia zbliżona do średniego poziomu w danym języku może oznaczać Cezara lub permutacyjny.
2) **Analiza histogramu** – jeśli można połączyć często występujące litery z ich częstotliwością występowania w danym języku może to oznaczać algorytm Cezara lub algorytm permutacyjny. Jeśli liter występujących jest mało, a pojawiają się często możemy mieć do czynienia z szyfrem adfgvx. Jeśli liter powtarzających się jest mało to możemy mierzyć się z Hillem lub Vigenerem.
3) **Analiza n-gramów** – jeśli niektóre fragmenty pojawiają się wyraźnie częściej niż pozostałe to najprawdopodobniej jest to szyfr cezara, adfgvx lub permutacyjny. Jeśli dużo różnych cząstek 3-4 znakowych możemy mieć do czynienia z Vigenerem lub Hillem.
4) **Analiza wykresu autokorelacji** – jeśli widać na wykresie widać powtarzalne w sposób okresowy fragmenty, to możemy mieć do czynienia z Vigenerem. Jeśli takiej powtarzalności brak – to może to być XOR.

Pytanie 2.10:

Analiza histogramu – jeśli mamy do czynienia z szyfrem Cezara lub Vigenera o krótkim kluczu, najczęściej występujące litery mogą odpowiadać literom o największej frekwencji w języku. Można na tej podstawie odtworzyć przesunięcia klucza.

Analiza n-gramów – można rozpoznać specyficzne wzorce, które mogły powstać w wyniku szyfrowania kluczami o określonej długości.

Analiza wykresu autokorelacji – pozwala wykryć powtarzające się wzorce w zaszyfrowanym tekście, co jest szczególnie przydatne dla szyfru Vinegera. Okresowe powtarzanie wzorców w tekście może wskazać klucz albo przynajmniej jego długość, co jest pierwszym krokiem do jego ustalenia.

Zad 3. Analiza dostarczonych plików.

Zadanie 3.1:

1_2.txt – Analiza histogramu, wskazuje na przesunięcie liter charakterystyczne dla szyfru Cezara. Najczęściej występującą literą była litera K, co sugerowało niewielkie przesunięcie względem standardowej częstotliwości liter w popularnych językach (gdzie najczęściej jest to A lub E). Niska entropia (4,10), która zbliżona jest do entropii popularnych języków również potwierdza teorię o wykorzystaniu szyfru Cezara. Korzystając z programu brute-force napisanego w python, i analizując przesunięcia na wszystkich 26 kombinacjach szyfru Cezara ustalono, że klucz to litera G (przesunięcie o 6).

```
Shift 0: ZNK KTOMSG SGINOTK CGY G LOKRJ ATOZ AYKJ OT CUXRJ CGX OO HE MKXSGT LOKRJ GMKTZY ZU KTIXEVZ GTJ JKIXEVZ SKYYGMKY GTJ I
Shift 1: YMJ JSNLRF RFHMNSJ BFX F KNJQI ZSNY ZXJI NS BTWQI BFW NN GD LJWRFS KNJQI FLJSYX YT JSHWDUY FSI IJHWDUY RJXXFLJX FSI H
Shift 2: XLI IRMKQE QEGLMRI AEW E JMIPH YRMX YWIH MR ASVPH AEV MM FC KIVQER JMIPH EKIRXW XS IRGVCTX ERH HIGVCTX QIWWEKIW ERH G
Shift 3: WKH HQLJPD PDFKLQH ZDV D ILHOG XQLW XVHG LQ ZRUOG ZDU LL EB JHUPDQ ILHOG DJHQWV WR HQFUBSW DQG GHFUBSW PHVVDJHV DQG F
Shift 4: VJG GPKIOC OCEJKPG YCU C HKGNF WPKV WUGF KP YQTNF YCT KK DA IGTOCP HKGNF CIGPVU VQ GPETARV CPF FGETARV OGUUCIGU CPF E
Shift 5: UIF FOJHNB NBDIJOF XBT B GJFME VOJU VTFE JO XPSME XBS JJ CZ HFSNBO GJFME BHFOUT UP FODSZQU BOE EFDSZQU NFTTBHFT BOE D
Shift 6: THE ENIGMA MACHINE WAS A FIELD UNIT USED IN WORLD WAR II BY GERMAN FIELD AGENTS TO ENCRYPT AND DECRYPT MESSAGES AND C
Shift 7: SGD DMHFLZ LZBGHMD VZR Z EHDKC TMHS TRDC HM VNQKC VZQ HH AX FDQLZM EHDKC ZFDMSR SN DMBQXOS ZMC CDBQXOS LDRRZFDR ZMC B
Shift 8: RFC CLGEKY KYAFGLC UYQ Y DGCJB SLGR SQCB GL UMPJB UYP GG ZW ECPKYL DGCJB YECLRQ RM CLAPWNR YLB BCAPWNR KCQQYECQ YLB A
Shift 9: QEB BKFDJX JXZEFKB TXP X CFBIA RKFQ RPBA FK TLOIA TXO FF YV DBOJXK CFBIA XDBKQP QL BKZOVMQ XKA ABZOVMQ JBPPXDBP XKA Z
Shift 10: PDA AJECIW IWYDEJA SWO W BEAHZ QJEP QOAZ EJ SKNHZ SWN EE XU CANIWJ BEAHZ WCAJPO PK AJYNULP WJZ ZAYNULP IAOOWCAO WJZ
Shift 11: OCZ ZIDBHV HVXCDIZ RVN V ADZGY PIDO PNZY DI RJMGY RVM DD WT BZMHVI ADZGY VBZION OJ ZIXMTKO VIY YZXMTKO HZNNVBZN VIY
Shift 12: NBY YHCAGU GUWBCHY QUM U ZCYFX OHCN OMYX CH QILFX QUL CC VS AYLGUH ZCYFX UAYHNM NI YHWLSJN UHX XYWLSJN GYMMUAYM UHX
Shift 13: MAX XGBZFT FTVABGX PTL T YBXEW NGBM NLXW BG PHKEW PTK BB UR ZXKFTG YBXEW TZXGML MH XGVKRIM TGW WXVKRIM FXLLTZXL TGW
Shift 14: LZW WFAYES ESUZAFW OSK S XAWDV MFAL MKWV AF OGJDV OSJ AA TQ YWJESF XAWDV SYWFLK LG WFUJQHL SFV VWUJQHL EWKKSYWK SFV
Shift 15: KYV VEZXDR DRTYZEV NRJ R WZVCU LEZK LJVU ZE NFICU NRI ZZ SP XVIDRE WZVCU RXVEKJ KF VETIPGK REU UVTIPGK DVJJRXVJ REU
Shift 16: JXU UDYWCQ CQSXYDU MQI Q VYUBT KDYJ KIUT YD MEHBT MQH YY RO WUHCQD VYUBT QWUDJI JE UDSHOFJ QDT TUSHOFJ CUIIQWUI QDT
Shift 17: IWT TCXVBP BPRWXCT LPH P UXTAS JCXI JHTS XC LDGAS LPG XX QN VTGBPC UXTAS PVTCIH ID TCRGNEI PCS STRGNEI BTHHPVTH PCS
Shift 18: HVS SBWUAO AOQVWBS KOG O TWSZR IBWH IGSR WB KCFZR KOF WW PM USFAOB TWSZR OUSBHG HC SBQFMDH OBR RSQFMDH ASGGOUSG OBR
Shift 19: GUR RAVTZN ZNPUVAR JNF N SVRYQ HAVG HFRQ VA JBEYQ JNE VV OL TREZNA SVRYQ NTRAGF GB RAPELCG NAQ QRPELCG ZRFFNTRF NAQ
Shift 20: FTQ QZUSYM YMOTUZQ IME M RUQXP GZUF GEQP UZ IADXP IMD UU NK SQDYMZ RUQXP MSQZFE FA QZODKBF MZP PQODKBF YQEEMSQE MZP
Shift 21: ESP PYTRXL XLNSTYP HLD L QTPWO FYTE FDPO TY HZCWO HLC TT MJ RPCXLY QTPWO LRPYED EZ PYNCJAE LYO OPNCJAE XPDDLRPD LYO
Shift 22: DRO OXSQWK WKMRSXO GKC K PSOVN EXSD ECON SX GYBVN GKB SS LI QOBWKX PSOVN KQOXDC DY OXMBIZD KXN NOMBIZD WOCCKQOC KXN
Shift 23: CQN NWRPVJ VJLQRWN FJB J ORNUM DWRC DBNM RW FXAUM FJA RR KH PNAVJW ORNUM JPNWCB CX NWLAHYC JWM MNLAHYC VNBBJPNB JWM
Shift 24: BPM MVQOUI UIKPQVM EIA I NQMTL CVQB CAML QV EWZTL EIZ QQ JG OMZUIV NQMTL IOMVBA BW MVKZGXB IVL LMKZGXB UMAAIOMA IVL
Shift 25: AOL LUPNTH THJOPUL DHZ H MPLSK BUPA BZLK PU DVYSK DHY PP IF NLYTHU MPLSK HNLUAZ AV LUJYFWA HUK KLJYFWA TLZZHNLZ HUK

Process finished with exit code 0
```

1_1.txt – Narzędzia autokorelacji w Cryptool wskazuje na to, że tekst został zaszyfrowany algorytmem powtarzającym klucz co stałą ilość znaków. Powtarzalne odległości luk między znakami wskazywały na długość klucza na około 6-8 znaków. Jest to cecha charakterystyczna między innymi szyfru Vinegera. Następnie wykorzystując tekst jawny z poprzedniego punktu, udało się przyporządkować wzrost liczbowy każdej z kolejnych liter (THEENIG… -> USYIFSE…), co pozwoliło ustalić, że klucz użyty do szyfrowania to „BLUESKY”.

1_3.txt – Analiza entropii i histogramu wykazała, że rozkład liter w zaszyfrowanym tekście jest identyczny w porównaniu do tekstu jawnego, jednak litery znajdują się na innych pozycjach. Litery bliżej końca alfabetu zdawały się pozostać bez zmian (przynajmniej niektóre). Sugerowało to użycie szyfru podstawieniowego. Skrypt w pythonie pozwolił porównać poszczególne litery z tekstu jawnego z literami w tekście zaszyfrowanym na tych samych pozycjach, co poskutkowało alfabetem: „QWERTYABCDFGHIJKLMNOPSUVXZ” -> co daje klucz „QWERTY”

```python
import string

def generate_substitution_key(plaintext, ciphertext):
    # utwórz listę dla klucza, wypełnioną początkowo znakami '-'
    alphabet = list(string.ascii_uppercase)
    key = ['-' for _ in range(26)]

    # przechodzimy przez tekst jawny i zaszyfrowany, mapując litery
    for pt_char, ct_char in zip(plaintext.upper(), ciphertext.upper()):
        if pt_char in alphabet and ct_char in alphabet:
            pt_index = alphabet.index(pt_char)
            key[pt_index] = ct_char

    # uzupełniamy brakujące litery w porządku rosnącym
    used_letters = set(key) - {'-'}
    remaining_letters = [ch for ch in alphabet if ch not in used_letters]

    # wstawiamy brakujące litery w odpowiednie miejsca
    for i in range(len(key)):
        if key[i] == '-':
            key[i] = remaining_letters.pop(0)

    return ''.join(key)

# przykładowe teksty jawny i zaszyfrowany
plaintext = """
The Enigma machine was a field unit used in World War II by German field ag
"""
ciphertext = """
Obt Ticahq hqebcit uqn q yctqr pico pntr ci Ujmqr Uqm CC wx Atmhqi yctqr qa
"""

# generowanie klucza
key = generate_substitution_key(plaintext, ciphertext)
print("Substitution key:", key)
```

```
Substitution key: QWERTYABCDFGHIJKLMNOPSUVXZ

Process finished with exit code 0
```

1_4.txt – Analiza zaszyfrowanego tekstu wskazuje na użycie algorytmu, który wyrównuje częstotliwości liter w histogramie. Wartość entropii na wysokim poziomie (4,52) oraz rozkład słupków na histogramie sugerują, że algorytm nieznacznie zmniejsza częstotliwość występowania najczęstszych liter i podnosi częstotliwość rzadszych. Pomimo tego nadal można zaobserwować obecność dwóch wyraźnych pików, prawdopodobnie odpowiadających literom W i H w szyfrogramie, które w tekście jawnym mogą odpowiadać literom A, E lub T.

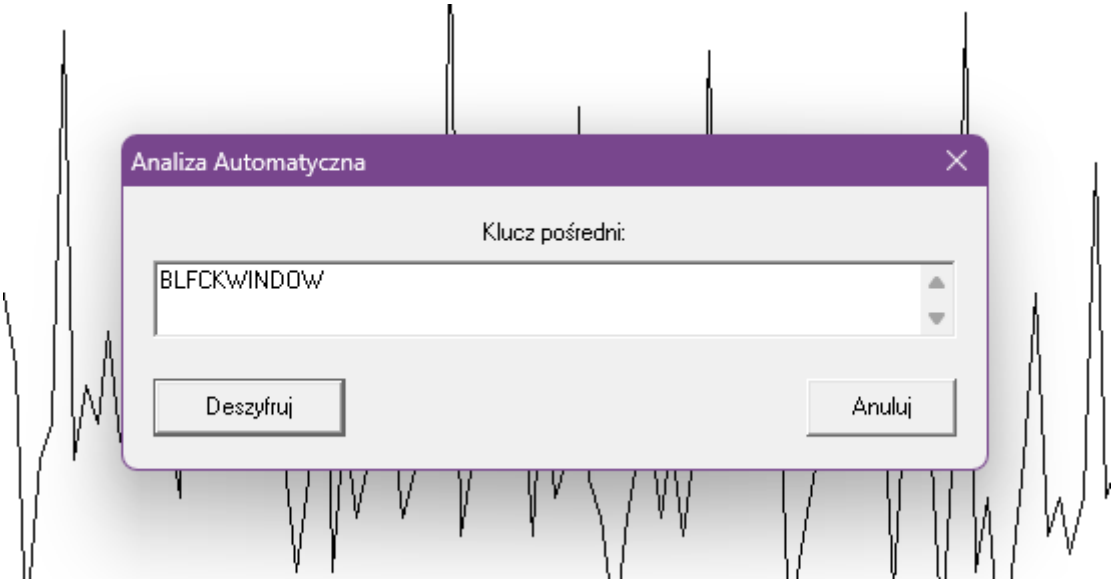
Autokorelacja nie wskazała na istnienie regularnych pików, co mogłoby sugerować prostą cykliczność w szyfrowaniu. Analiza ngramów wykazała spłaszczenie rozkładu – zmniejszyła się częstotliwość występowania charakterystycznych dla języka angielskiego ngramów, a w zamian pojawiło się więcej różnorodnych ngramów o niższej częstotliwości.

Chociaż nie udało mi się zidentyfikować klucza szyfrowania, zauważyłem pewne regularności w postaci liter, które często pozostają bez zmian w szyfrogramie (np. A->A, N->N) na tych samych pozycjach). Na podstawie tych obserwacji napisałem skrypt w Pythonie, który wygenerował potencjalny alfabet-klucz: „AZYXWVUTS_QPONMLYJIHGFEDCB”. Niestety, ten klucz nie pasuje do żadnego znanego algorytmu szyfrującego i nie rozwiązuje problemu.

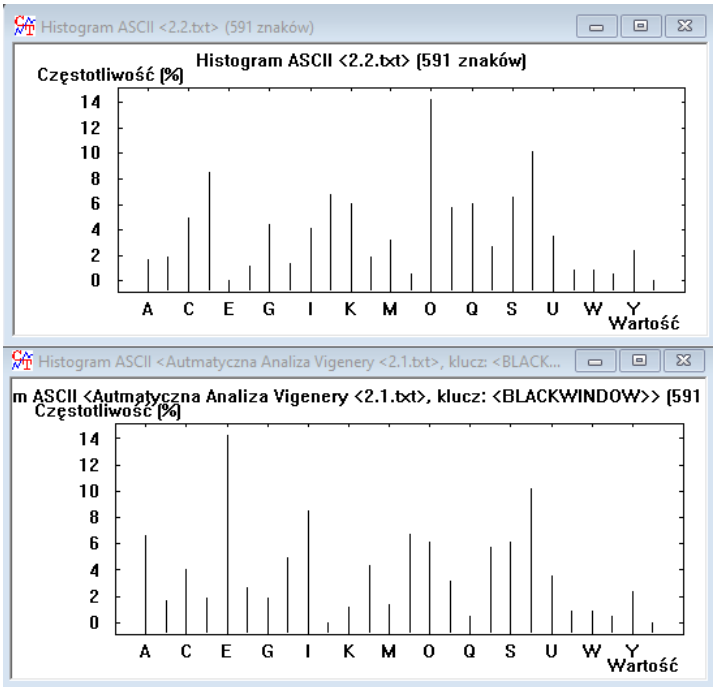
Letter	Mapping	Count	PlainCount	Effectiveness
A	A	41	89	0.46
B	Z	8	12	0.67
C	Y	29	42	0.69
D	X	21	41	0.51
E	W	67	150	0.45
F	V	11	19	0.58
G	U	9	22	0.41
H	T	36	61	0.59
I	S	44	82	0.54
J	_	0	0	0.00
K	Q	4	5	0.80
L	P	28	46	0.61
M	O	12	27	0.44
N	N	44	74	0.59
O	M	35	66	0.53
P	L	12	27	0.44
Q	Y	1	1	1.00
R	J	33	70	0.47
S	I	28	55	0.51
T	H	64	119	0.54
U	G	12	26	0.46
V	F	2	2	1.00
W	E	17	30	0.57
X	D	2	2	1.00
Y	C	12	19	0.63
Z	B	1	1	1.00

Zadanie 3.2:

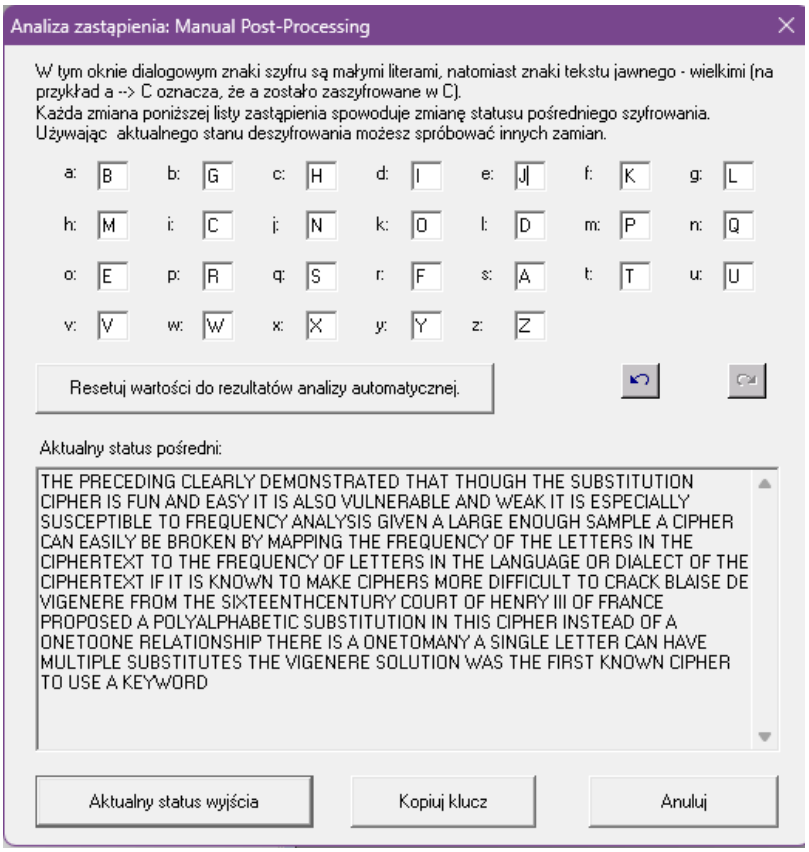
2_1.txt – autokorelacja powtarzająca się co określony okres czasu, więc podejrzewam vinegera o długości klucza około 11. Używając dostępnych opcji automatycznego łamania szyfru dla szyfru vinegera analiza automatyczna wykryła klucz pośredni: „BLFCKWINDOW” co intuicyjnie poprawiłem na „BLACKWINDOW”.



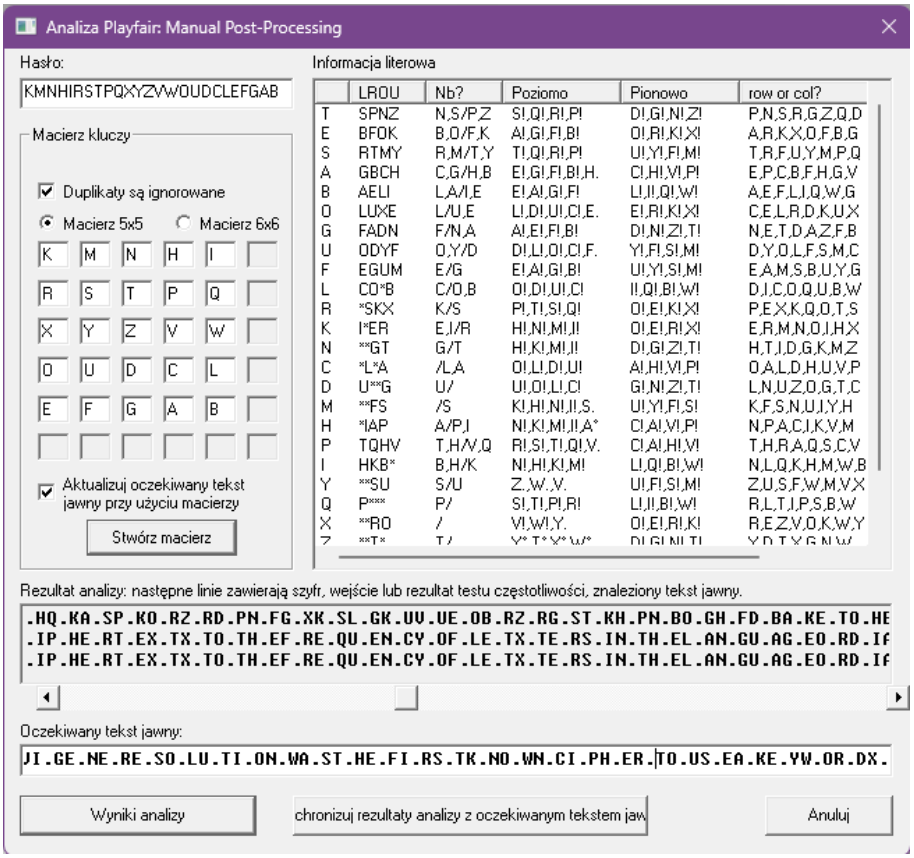
2_2.txt – Niska entropia (4,15) może sugerować szyfr atbash lub permutacyjny. Końcowa część histogramu pokrywa się z histogramem tekstu jawnego, co wskazuje na szyfr atbash. Wysokości słupków na histogramie mają te same piki i dołki.



Po zabawie w analizie ręcznej szyfru zastąpienia, udało się odnaleźć klucz -> „SAILOR”



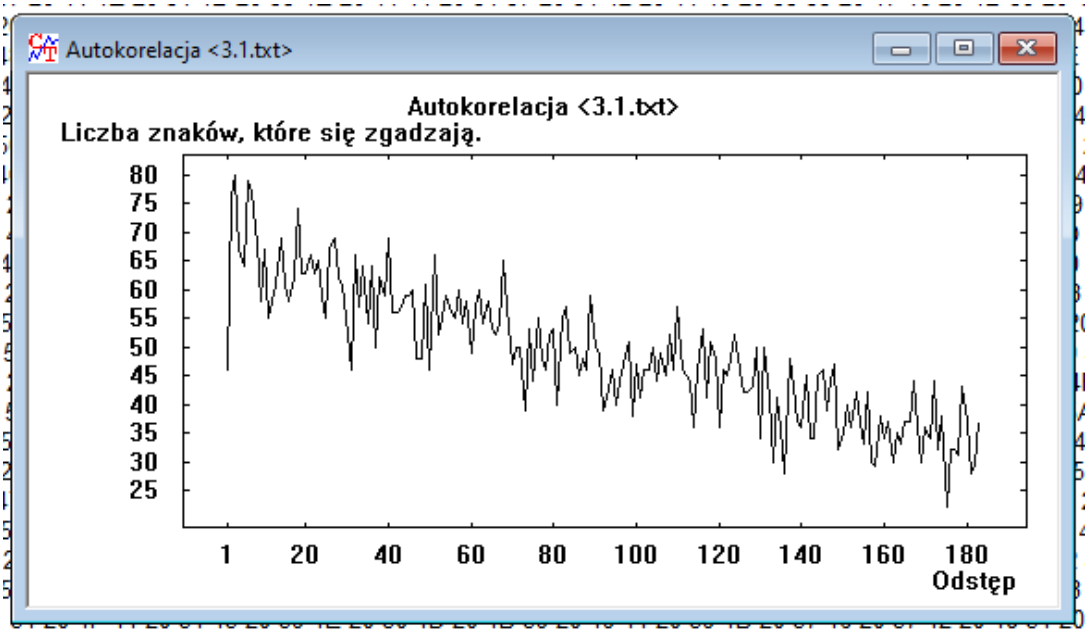
2_3.txt – Format zaszyfrowanego tekstu sugeruje użycie algorytmu Playfair. Entropia (4,46). Używając ręcznej analizy playfair doszedłem tylko do tego momentu:



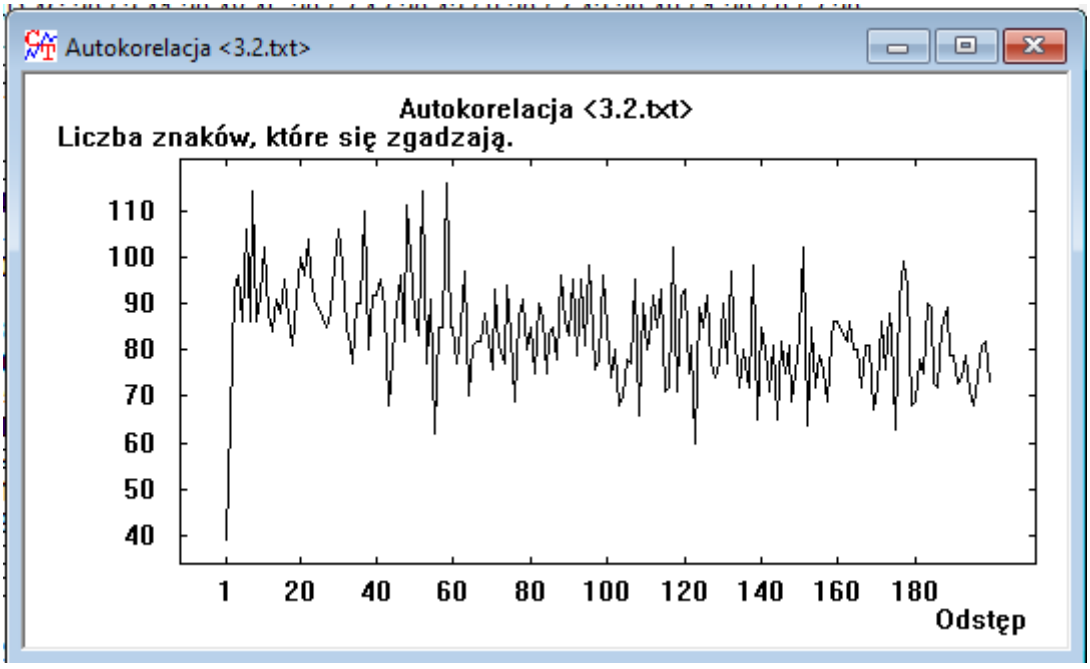
Jednak klucz: „KMNHIRSTPQXYZVWOUDCLEFGAB” nic mi nie mówi, więc nie udało mi się dokładnie złamać tego kodu. Jestem jednak pewien że byłem blisko, bo przy wpisywaniu tekstu jawnego, algorytm podpowiadał mi dalszą jego część poprawnie bez jego znajomości, więc sam klucz gdzieś musiał być tylko nie umiałem skorzystać poprawnie z narzędzia.

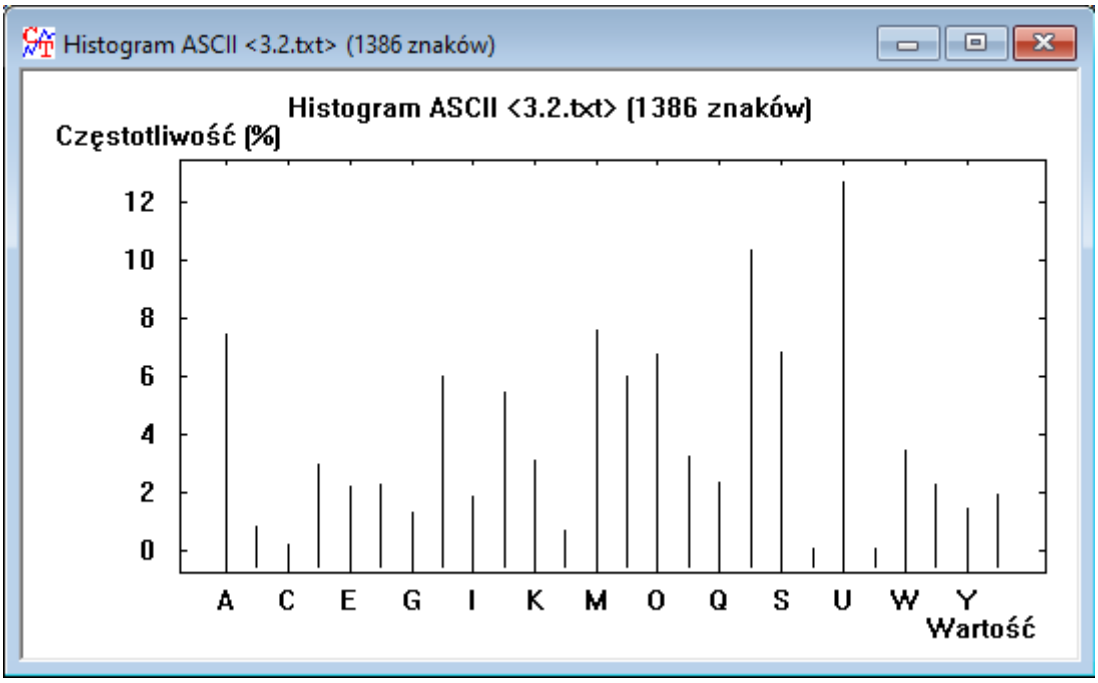
Zadanie 3.3:

3_1.txt – Hexadecymalne kodowanie sugeruje wykorzystanie algorytmu XOR lub Homofonów. Autokorelacja ma trend spadkowy. Nic więcej nie udało mi się ustalić.

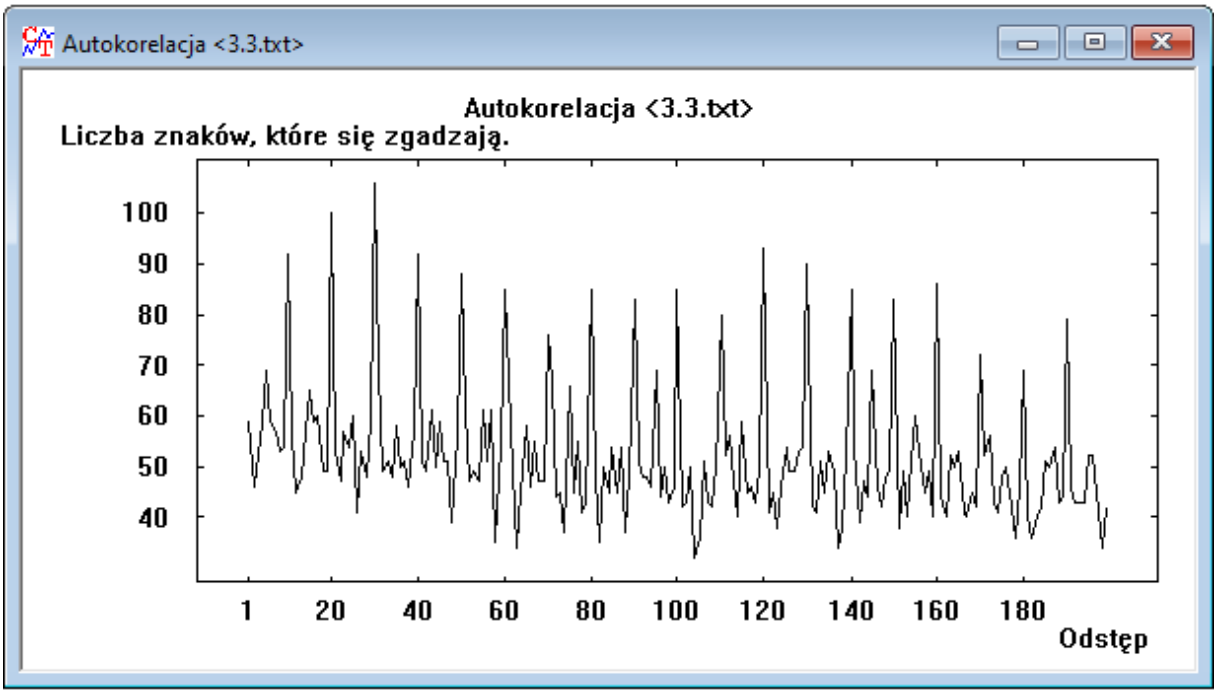


3_2.txt – Entropia (4,20) może wskazywać że nie jest to ani szyfr Hilla czy Vinegera, gdyż byłaby ona wyższa. Na wykresie autokorelacji nie widać znaczącej cykliczności pików co wyklucza wykorzystanie Vinegera. Histogram sugeruje, że litery ‘U’ oraz ‘R’ mogą odpowiadać popularnym literom w języku polskim lub angielskim (np. ‘A’ lub ‘E’). Na pewno nie jest to szyfr Cezara.





3_3.txt – Bardzo wysoka entropia (4,63) wskazuje na to, że jest to Hill lub Vineger. Cykliczna autokorelacja pozwala zauważyć że jest to Vineger, o długości klucza około 10 znaków. Wykorzystując automatyczne narzędzia do analizy szyfru Vinegera, udało się odnaleźć klucz „WHITESTONE”.



Pytanie 3.4:

Siła algorytmu szyfrującego zależy od kilku czynników:

- Długość i złożoność klucza, im dłuższy i bardziej złożony klucz, tym trudniej go złamać. Krótkie klucze są podatne na ataki brute-force, a klucze mniej złożone możemy łatwiej rozpoznać kiedy się wyłonią (jeśli są cokolwiek znaczącym ciągiem znaków). Gdy klucz jest długi i zawiera znaki trudniejsze do zgadnięcia, staje się on bezpieczniejszy.
- Wysokość entropii, im większa entropia tym ciężej odnaleźć wzorce np. na histogramie, ponieważ tekst jest mniej przewidywalny i trudniejszy do odgadnięcia. Szyfr nie powinien ujawniać żadnych przewidywalnych wzorców.
- Istotne jest też, aby algorytm ukrywał naturę tekstu i ewentualną cykliczność na autokorelacji, nie powinien też ujawniać czy tekst zawiera dużo powtarzających się znaków, czy nie.

Pytanie 3.5:

Dla niektórych algorytmów wielokrotne szyfrowanie może zwiększyć siłę szyfrowania (np. szyfr Hilla). Jednak na ogół lepiej zwiększyć złożoność i długość klucza, uniknąć w nim powtórzeń i cykliczności samego klucza. Warto też korzystać z elementu losowości, tak aby nie dało się znaleźć w naszym zaszyfrowanym tekście powtarzalnych wzorców np. przez wybór dość losowego algorytmu (przykładowo w algorytmie homofonicznym, homofony są generowane w sposób losowy, co utrudnia jego złamanie), albo losowy dobór klucza.