

SPRAWOZDANIE

Algorytmy i Struktury Danych

Programowanie dynamiczne

Bartłomiej Kowalewski, 145204, I5.2

Jakub Koza, 145436, I5.2

Pomiary wykonane zostały na sprzęcie:

Laptop Lenovo G710

Windows 10 Home 64-bit

Procesor: Intel Core i7-4702MQ (2.2-3.2 GHz)

Pamięć RAM: 16 GB

Dysk twardy: 1TB SSHD

Karta graficzna: Intel HD Graphics 4600

Algorytmy zaimplementowane zostały w języku Python 3.7:

W całości sprawozdania czas podawany jest w mikrosekundach. Na wykresach pominęliśmy zaznaczanie odchylenia standardowego, ze względu na jego bardzo małe wartości. Pomiary czasu wykonane zostały za pomocą biblioteki datetime.

1. Zależność czasu wykonywania obliczeń od liczby elementów dla stałej pojemności plecaka obrazują następujące tabele:

Zależność czasu wykonywania APD od liczby elementów					
n	1. pomiar	2. pomiar	3. pomiar	4. pomiar	5. pomiar
100	124955	124962	116933	130927	107062
200	250006	237915	210906	228899	236983
300	359347	386747	317862	321439	325933
400	484368	523656	471743	458705	448808
500	624917	581663	565390	567750	593142
600	765567	732680	674594	664253	652797
700	921775	915805	766777	833830	786497
800	1031174	946994	903720	966103	919450
900	1109287	1036402	1065203	1028578	1067354
1000	1328026	1103152	1111717	1122353	1222022

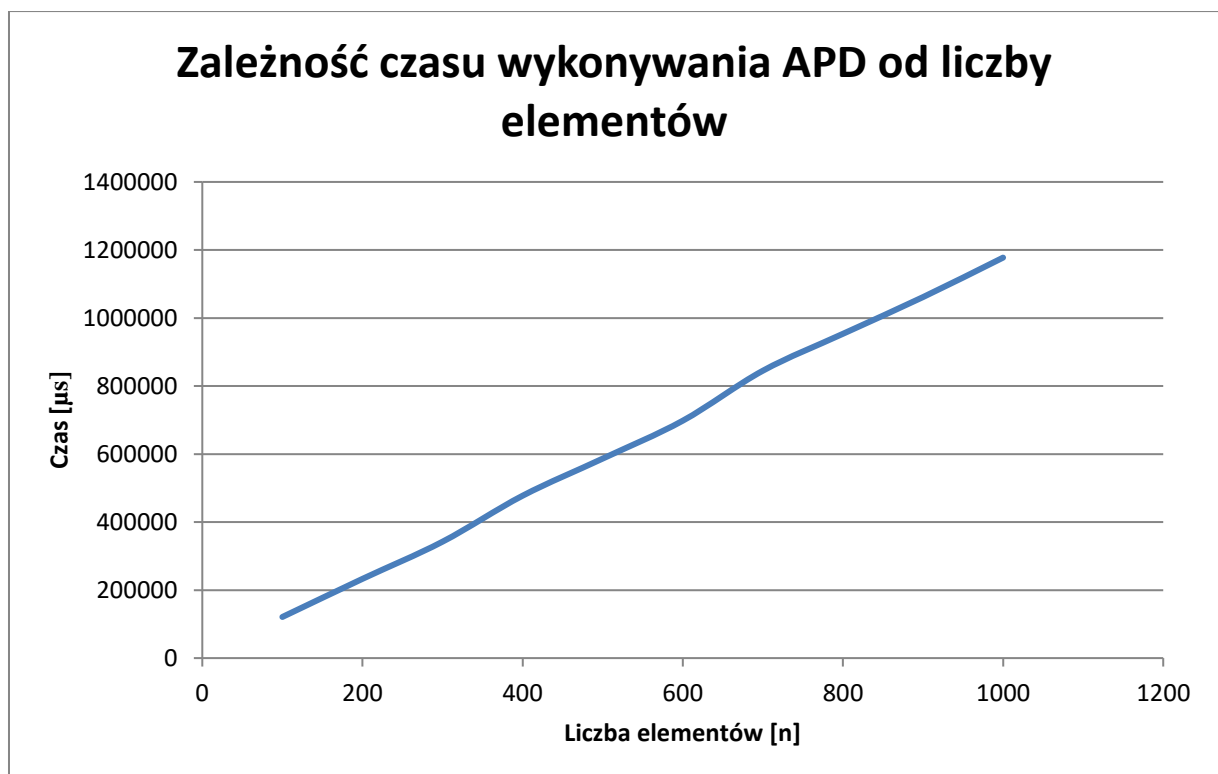
Zależność czasu wykonywania Algorytmu Wyczerpującego od liczby elementów					
n	1. pomiar	2. pomiar	3. pomiar	4. pomiar	5. pomiar
10	62495	49967	104001	48369	65962
11	109367	108938	233884	131307	116967
12	234391	267846	454057	271812	212902
13	499936	516700	1006005	530729	469694
14	1031202	1155255	2124933	1183291	976816
15	2249833	2253626	4619933	2313693	2078998
16	4905890	4908355	9690909	4557338	4356115
17	10686687	10291147	20969061	9595568	9614078
18	21920239	21837277	61960116	20953186	21110781
19	46105999	63611510	94734581	62518324	61447680

Zależność czasu wykonywania Algorytmu Zachłannego od liczby elementów

n	1. pomiar	2. pomiar	3. pomiar	4. pomiar	5.pomiar
100	0	6998	6995	6995	7996
200	15623	28954	30979	28982	29944
300	78155	87081	82965	62962	67988
400	140584	143513	105985	115003	129012
500	187517	166090	167996	170964	172093
600	296818	266981	236891	251149	267033
700	406230	326059	317889	343805	343451
800	499965	461734	432804	474899	519736
900	640610	589755	576801	566429	563746
1000	796815	710743	770202	728844	726686

a) Algorytm programowania dynamicznego ($b = 200$)

Zależność czasu wykonywania APD od liczby elementów		
n	średnia	odchylenie
100	120968	8255
200	232942	12916
300	342266	26707
400	477456	26028
500	586572	21636
600	697978	43584
700	844937	64147
800	953488	44434
900	1061365	28444
1000	1177454	86645



OBSERWACJE: Czas wykonywania algorytmu programowania dynamicznego dla problemu plecakowego w zależności od liczby elementów rośnie liniowo.

b) Algorytm Zachłanny ($b = 200$)

Zależność czasu wykonywania Algorytmu Zachłannego od liczby elementów		
n	średnia	odchylenie
100	5797	2924
200	26896	5686
300	75830	9055
400	126819	14468
500	172932	7596
600	263774	19974
700	347487	31035
800	477828	30137
900	587468	28101
1000	746658	31863



OBSERWACJE: Czas wykonywania algorytmu zachłannego dla problemu plecakowego w zależności od liczby elementów rośnie liniowo.

c) Algorytm Wyczerpujący ($b = 100$)

Zależność czasu wykonywania Algorytmu Wyczerpującego od liczby elementów		
n	średnia	odchylenie
10	66159	20120
11	140093	47589
12	288202	85740
13	604613	201724
14	1294299	422285
15	2703217	961547
16	5683721	2014698
17	12231308	4388517
18	29556320	16206421
19	65683619	15871342



OBSERWACJE: Czas wykonywania algorytmu wyczerpującego dla problemu plecakowego w zależności od liczby elementów rośnie wykładniczo.

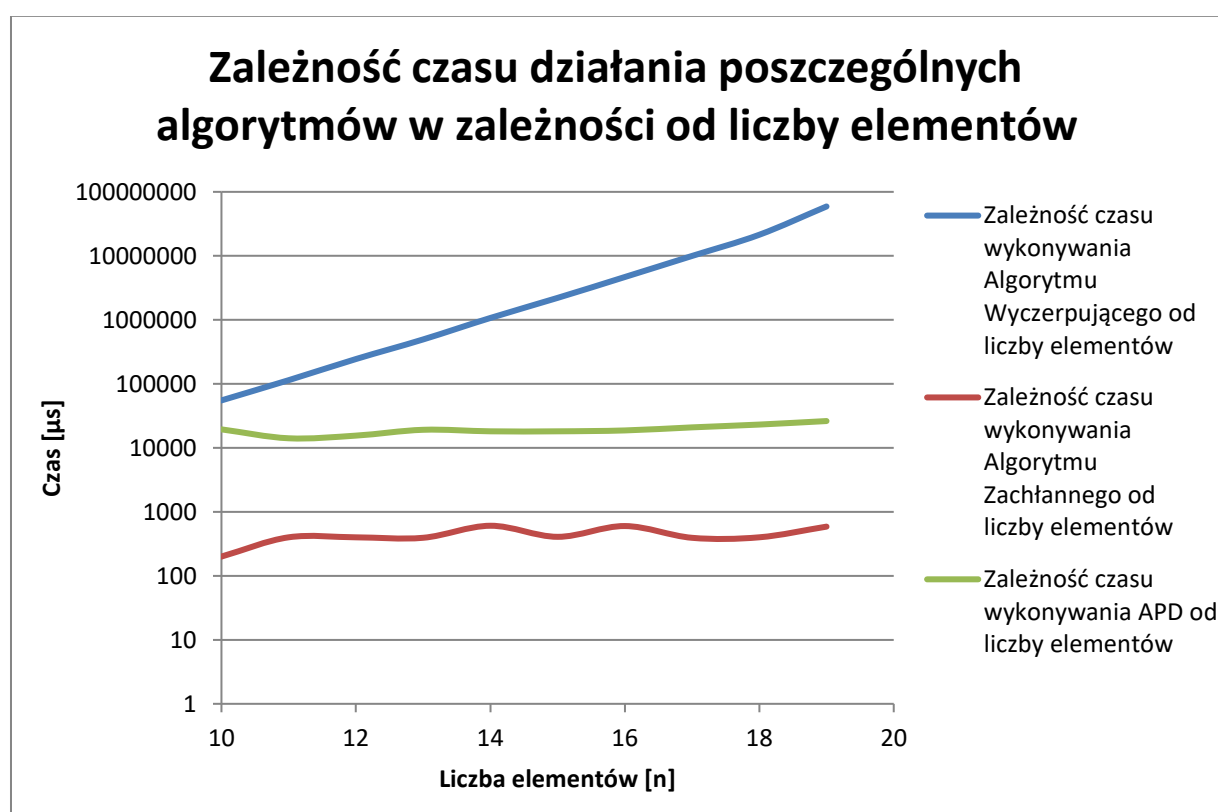
2. Zależność czasu wykonywania algorytmów w zależności od liczby elementów w skali logarytmicznej.

Zależność czasu wykonywania Algorytmu Wyczerpującego od liczby elementów					
n	1. pomiar	2. pomiar	3. pomiar	4. pomiar	5. pomiar
10	62495	49967	48971	48369	65962
11	109367	108938	104001	131307	116967
12	234391	267846	233884	271812	212902
13	499936	516700	454057	530729	469694
14	1031202	1155255	1006005	1183291	976816
15	2249833	2253626	2124933	2313693	2078998
16	4905890	4908355	4619933	4557338	4356115
17	10686687	10291147	9690909	9595568	9614078
18	21920239	21837277	20969061	20953186	21110781
19	46105999	63611510	61960116	62518324	61447680

Zależność czasu wykonywania Algorytmu Zachłannego od liczby elementów					
n	1. pomiar	2. pomiar	3. pomiar	4. pomiar	5. pomiar
10	0	0	0	1000	0
11	0	0	0	1000	999
12	0	1000	0	0	998
13	0	965	0	0	1000
14	0	998	1035	0	998
15	0	999	0	0	1035
16	0	998	999	0	1000
17	0	964	0	999	0
18	0	1000	0	999	0
19	0	0	974	1000	963

Zależność czasu wykonywania APD od liczby elementów					
n	1. pomiar	2. pomiar	3. pomiar	4. pomiar	5. pomiar
10	31244	15986	17991	15980	15989
11	14556	11999	15984	12992	14996
12	15623	15026	16025	14990	15980
13	31246	17470	14992	16991	14987
14	15622	19997	17993	16032	20992
15	15623	17015	16989	17989	22982
16	15624	17982	16989	20957	21977
17	15586	18957	20951	25993	22992
18	15589	21020	28986	23982	25976
19	31246	21952	26012	29978	21985

Zależność czasu wykonywania Algorytmu Wyczerpującego od liczby elementów						
n	Alg. Wyczerpujący		Alg. Zachłanny		APD	
	średnia	odchylenie	średnia	odchylenie	średnia	odchylenie
10	55153	7508	40828	24602	32158	28777
11	114116	9543	88072	51751	60159	55494
12	244167	22377	179653	105843	119865	109516
13	494223	28585	363623	211876	248129	234209
14	1070514	82929	791532	463763	532568	493085
15	2204217	87719	1629410	944834	1092737	1050262
16	4669526	212687	3383351	1955799	2309586	2207300
17	9975678	438494	7225143	4171582	5007067	4792026
18	21358109	429438	15758262	9098233	10724586	10511319
19	59128726	6550885	46481535	26838787	31781827	28612058



OBSERWACJE: Przy stałej pojemności plecaka czas wykonywania algorytmu wyczerpującego jest najdłuższy, znacznie krótszy okazuje się czas wykonywania algorytmu zachłanny natomiast czas wykonywania algorytmu programowania dynamicznego plasuje się pomiędzy czasami powyżej wymienionych algorytmów.

3. Zależność czasu wykonywania obliczeń od pojemności plecaka dla stałej liczby elementów przedstawiają następujące tabele

Zależność czasu wykonywania APD od pojemności plecaka					
b	1. pomiar	2. pomiar	3. pomiar	4. pomiar	5. pomiar
50	124991	108013	123920	122968	104976
100	296817	264846	275859	261834	254892
150	453128	432003	453255	408000	403732
200	640611	588096	550773	560592	555503
250	796851	751584	716346	712550	695635
300	953018	959567	850924	867044	891076
350	1140577	1204658	1039671	994675	991414
400	1343650	1250962	1247424	1191287	1192641
450	1484265	1409770	1427257	1399181	1390924
500	1671778	1499095	1528531	1513435	1534555

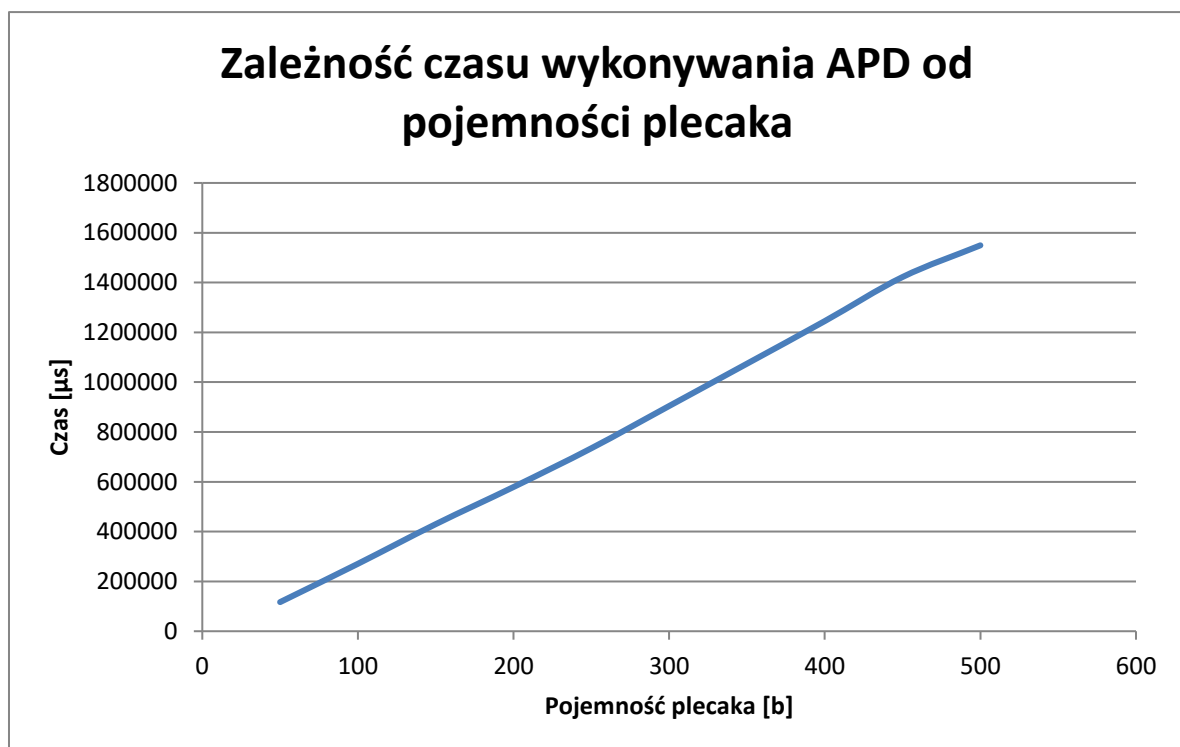
Zależność czasu wykonywania Algorytmu Wyczerpującego od pojemności plecaka					
b	1. pomiar	2. pomiar	3. pomiar	4. pomiar	5. pomiar
20	31240	56021	56735	46007	43010
30	62496	44072	47972	42022	41976
40	46906	46035	42996	41976	41942
50	46870	56992	44974	43056	43938
60	46841	61000	42939	42974	44012
70	62493	47971	44976	71436	44937
80	46881	50023	50969	45974	45975
90	46871	66009	54997	44021	44971
100	62487	50998	47972	44975	44973
110	78122	55020	49940	45053	45972

Zależność czasu wykonywania Algorytmu Zachłannego od pojemności plecaka

b	1. pomiar	2. pomiar	3. pomiar	4. pomiar	5.pomiar
50	187486	165928	175928	185957	177628
100	203111	210219	174949	189985	190889
150	203110	179938	168978	199947	187856
200	187450	173925	193911	188952	191883
250	187522	178927	196421	185945	209879
300	203111	211896	185982	171989	185910
350	187487	186993	168938	220885	199994
400	203085	185013	184991	171992	174654
450	203136	166230	185076	173909	203887
500	203078	183892	177517	196923	179890

a) Algorytm Programowania Dynamicznego (n = 500)

Zależność czasu wykonywania APD od pojemności plecaka		
b	średnia	odchylenie
50	116974	8634
100	270850	14638
150	430024	21231
200	579115	33366
250	734593	36059
300	904326	44362
350	1074199	84611
400	1245193	55497
450	1422279	33287
500	1549479	62380



OBSERWACJE: Czas wykonywania algorytmu programowania dynamicznego w zależności od pojemności plecaka wydłuża się liniowo wraz z rosnącą pojemnością.

b) Algorytm Zachłanny (n = 500)

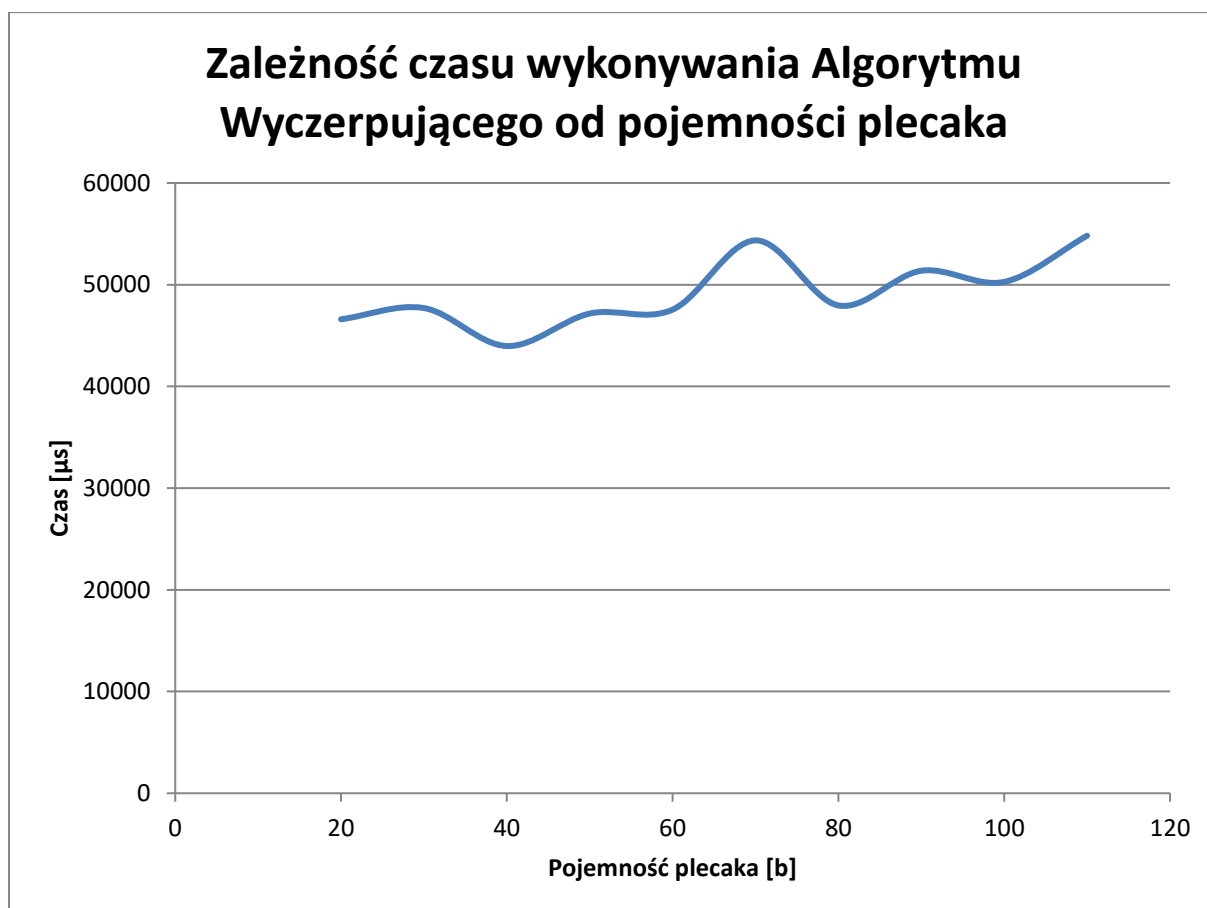
Zależność czasu wykonywania Algorytmu Zachłannego od pojemności plecaka		
b	średnia	odchylenie
50	178585	7769
100	193831	12122
150	187966	12632
200	187224	7019
250	191739	10644
300	191778	14091
350	192859	17157
400	183947	10934
450	186448	15169
500	188260	9992



OBSERWACJE: Czas wykonywania algorytmu zachłannego w zależności od pojemności plecaka jest stały.

c) Algorytm Wyczerpujący (n = 10)

Zależność czasu wykonywania Algorytmu Wyczerpującego od pojemności plecaka		
b	średnia	odchylenie
20	46603	9388
30	47708	7709
40	43971	2094
50	47166	5075
60	47553	6872
70	54363	10728
80	47964	2115
90	51374	8278
100	50281	6499
110	54821	12170

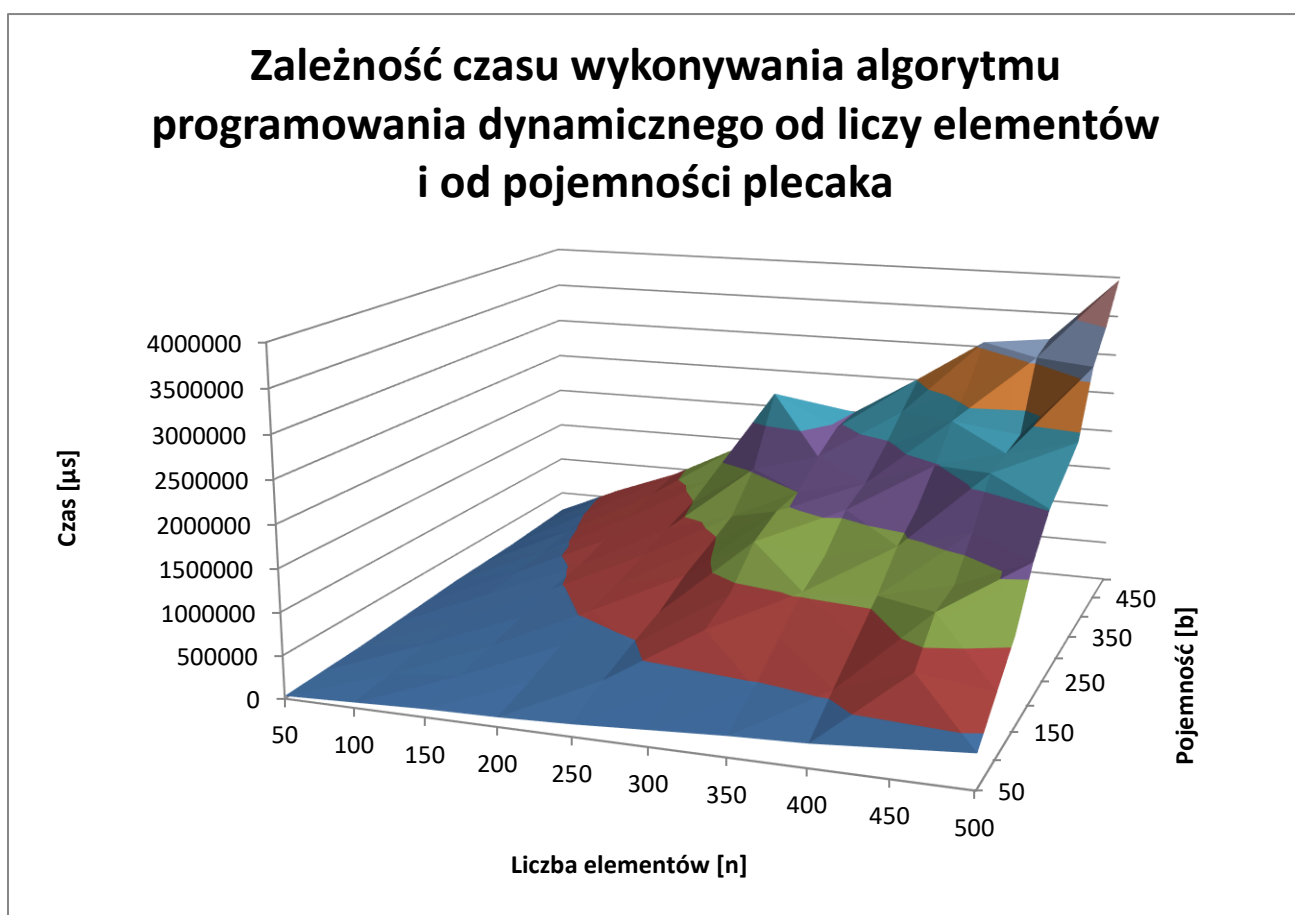


OBSERWACJE: Czas wykonywania algorytmu wyczerpującego w zależności od pojemności plecaka jest stały.

4. Zależność czasu wykonywania algorytmów od liczby elementów i od pojemności plecaka

a) Algorytm Programowania Dynamicznego

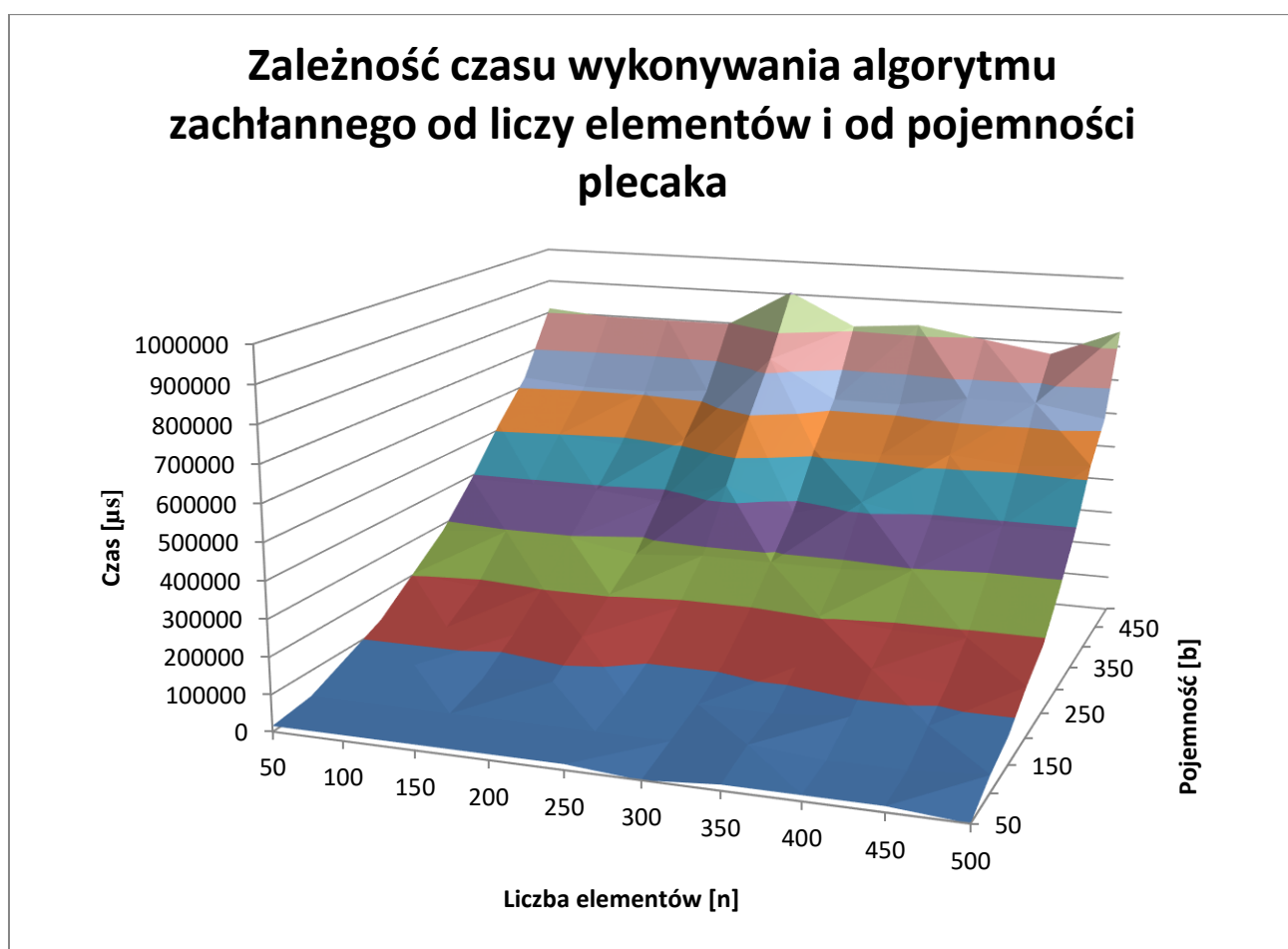
Zależność czasu wykonywania algorytmu programowania dynamicznego od liczby elementów i od pojemności plecaka										
	50	100	150	200	250	300	350	400	450	500
50	31248	62525	93772	109402	140615	187519	234364	265604	328128	390592
100	46872	109333	187487	234357	343724	406218	468709	546831	843654	734319
150	62459	187521	281194	374972	484346	749948	749902	812440	1234287	1078046
200	93778	234332	374998	531177	609362	921774	1031174	1077978	1281157	1640511
250	124992	312472	468716	624959	812444	1312410	1218625	1421777	1671749	2015470
300	156206	359345	531212	749981	953084	1468676	1562387	1640508	2156122	2202993
350	171861	406215	640577	874900	1109296	1499863	1687379	1984233	2202929	2406066
400	187455	468742	734320	984339	1531102	1765464	1937357	2359207	2249859	3156015
450	218764	546836	843688	1124881	2359203	1968612	2234211	2624811	3062282	3515367
500	249972	609300	921807	1343651	1968606	1999853	2515476	3062278	3156025	3968458



OBSERWACJE: Czas wykonywania algorytmu programowania dynamicznego rośnie wraz ze wzrostem liczby elementów oraz wzrostem pojemności plecaka.

b) Algorytm Zachłanny

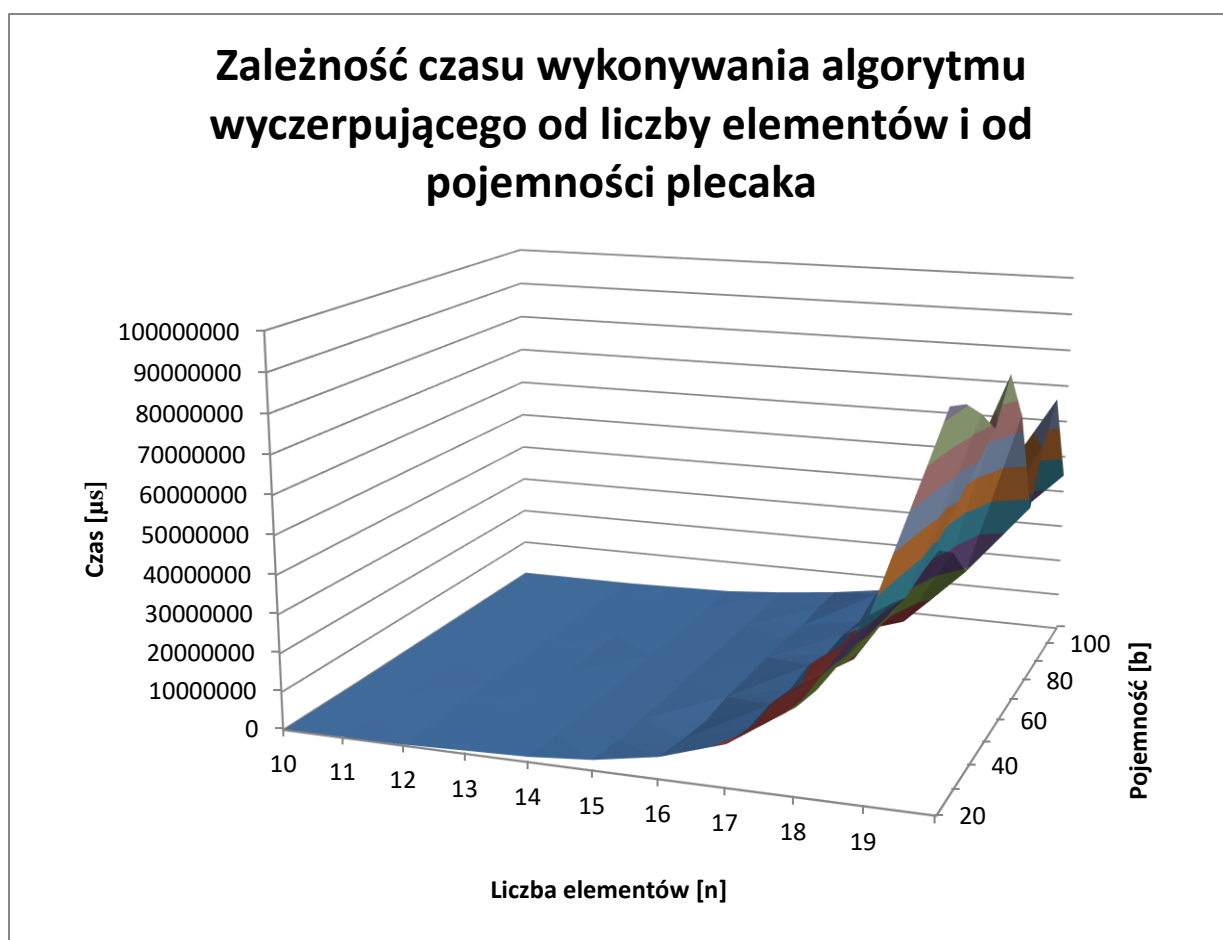
Zależność czasu wykonywania algorytmu zachłannego od liczby elementów i od pojemności plecaka										
	50	100	150	200	250	300	350	400	450	500
50	15590	15624	15624	15594	15628	0	15625	15652	15624	0
100	31281	31246	31249	31248	31245	31248	46876	31214	31249	46878
150	78087	78120	62467	78114	62521	62494	62525	78153	78090	78120
200	124988	124963	124988	140615	109367	109404	124960	140615	125019	140642
250	203106	187514	203143	203146	187450	187514	203111	187487	187452	187454
300	281226	296851	296852	265604	281228	281228	281261	296852	281235	281261
350	390597	390596	390597	374964	437469	374972	421812	390562	390627	390597
400	515592	499928	484341	515624	593742	515551	515588	531246	515588	531212
450	624953	609363	609331	624953	734285	624988	624953	656202	656200	624953
500	812441	796816	796815	796785	906188	812440	828036	796816	765569	843687



OBSERWACJE: Czas wykonywania algorytmu zachłannego rośnie wraz ze wzrostem pojemności plecaka.

c) Algorytm Wyczerpujący

Zależność czasu wykonywania algorytmu wyczerpującego od liczby elementów i od pojemności plecaka										
	20	30	40	50	60	70	80	90	100	110
10	46880	46875	46873	31244	62504	46878	46869	62462	46872	46836
11	171889	124960	109367	109363	109329	124991	109365	93739	93769	109364
12	312445	218697	203110	249982	218728	234351	203071	203107	218730	218733
13	843688	484312	421844	531206	468742	437439	437433	437494	453091	578048
14	1359282	937432	953055	984340	968678	1218628	968712	921804	953050	953028
15	2718542	1937353	1937357	1952981	1999820	2140465	1999848	1890511	1937382	2249830
16	5608972	4171596	3156019	4249654	4202791	4484042	4249715	3530976	4218406	3984079
17	10921070	9218036	9702415	8093195	9499300	7233827	7733834	6499514	9577421	6374493
18	21982763	20701607	21029712	18108013	20139149	18420519	13061507	15561319	19451663	16623763
19	45543547	43059344	40215774	41996922	43028094	38044048	28544780	36747260	43512436	28997904



OBSERWACJE: Czas wykonywania algorytmu wyczerpującego rośnie wraz ze wzrostem liczby elementów.

5. Wnioski

Złożoność obliczeniowa algorytmów

Złożoność obliczeniowa	
Algorytm Zachłanny	$O(n \log_2 n)$
Algorytm Wyczerpujący	$O(2^n)$
APD	$O(n \times c)$

Klasa złożoności

Binarny problem plecakowy (klasa złożoności problemu)	
Problem optymalizacyjny	NP-trudny
Problem decyzyjny	NP-zupełny

W przypadku poszukiwania rozwiązania dla zwiększającej się liczby elementów w zbiorze zauważamy, zarówno w przypadku algorytmu zachłannego, jak i APD, liniowy wzrost czasu w porównaniu do liczby elementów. W przypadku algorytmu zachłannego potrzebujemy jedynie posortować dane wejściowe i przeiterować się po elementach sprawdzając czy mieszczą się do plecaka. Algorytm programowania dynamicznego tworzy macierz kosztów, i aby uzyskać odpowiedź, wystarczy wykonać tyle przejść w pętli ile znajduje się przedmiotów w zbiorze. Sytuacja wygląda inaczej w przypadku algorytmu wyczerpującego. Tutaj w celu znalezienia odpowiedzi potrzebne jest stworzenie tablicy wszystkich kombinacji zbioru więc 2^n przypadków i każdy z nich może być rozwiązaniem. Dlatego też nawet dla stosunkowo niewielkich ilości przedmiotów w zbiorze czas wykonywania algorytmu okazuje się bardzo długi i wykres rośnie wykładniczo.

W przypadku zależności czasu wykonywania algorytmu od objętości plecaka zmiany można zauważyć jedynie w przypadku algorytmu programowania dynamicznego. Tutaj wykres rośnie linowo z powodu potrzeby wypełnienia większej ilości komórek w macierzy wyników. W przypadku algorytmu wyczerpującego nie zmienia się ilość wykonanych iteracji, niezależnie od pojemności plecaka.

Warto zaznaczyć, że algorytm zachłanny nie zawsze daje rozwiązanie optymalne mimo szybkiego czasu działania. Dzieje się tak z powodu z braku zachowania przez problem pewnych własności: optymalnej podstruktury oraz wyboru zachłannego. Pierwsza z podanych mówi o tym, że najlepsze rozwiązanie problemu jest funkcją optymalnych rozwiązań podproblemów tzn. znając rozwiązania podproblemów, możemy wyznaczyć rozwiązanie problemu głównego. Kolejną wspomnianą cechą, przy której algorytm zachłanny może dać rozwiązanie optymalne jest własność wyboru zachłannego, która oznacza, że za pomocą lokalnie optymalnych wyborów można znaleźć rozwiązanie globalnie optymalne (wystarczy rozwiązać tylko ten podproblem, który wydaje się dać rozwiązanie optymalne dla głównego problemu). Gdy problem nie spełnia dwóch powyżej opisanych własności, wówczas możemy sądzić, że prawdopodobnie nie uzyskamy optymalnego rozwiązania.