

Sprawozdanie

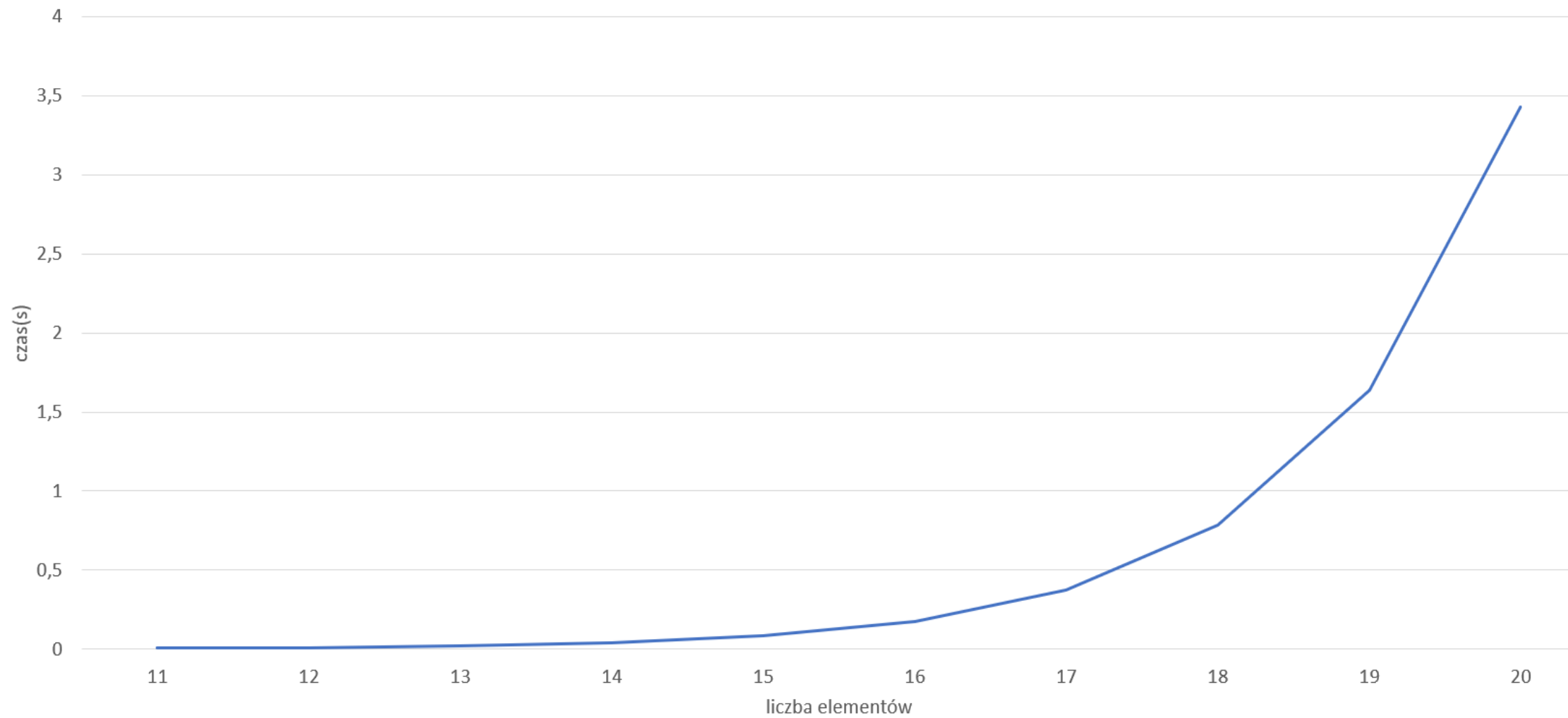
Programowanie dynamiczne

Mikołaj Pluta

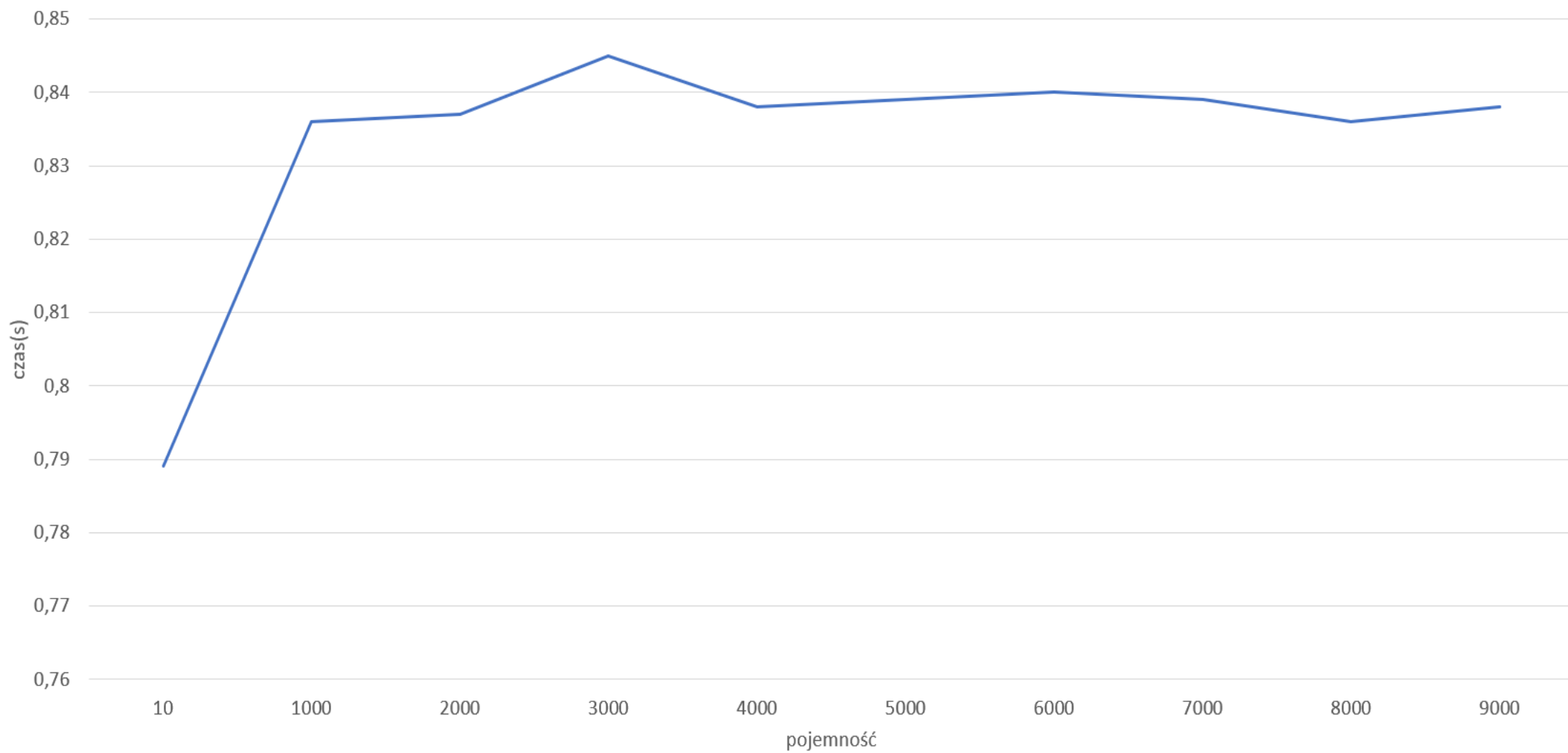
- 1) Celem ćwiczenia jest porównanie działania trzech algorytmów rozwiązujących problem „plecakowy”. Całość została zaimplementowana w języku Python.

2) Algorytm siłowy.

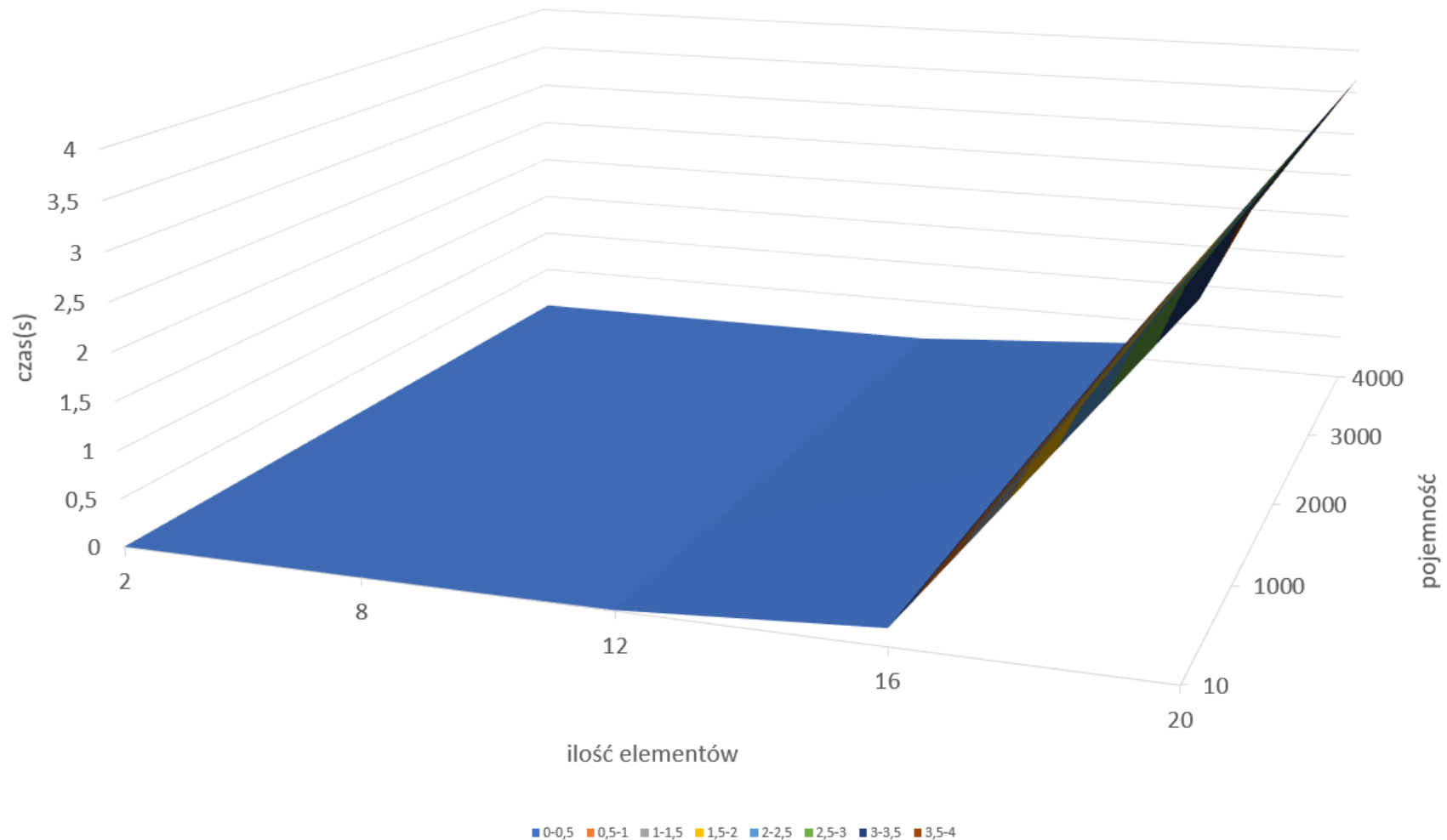
2.1) Czas wykonywania algorytmu siłowego w zależności od liczby elementów (stała pojemność $c = 10$).



2.2)Czas wykonywania algorytmu siłowego w zależności od pojemności(stała liczba elementów $n = 18$).



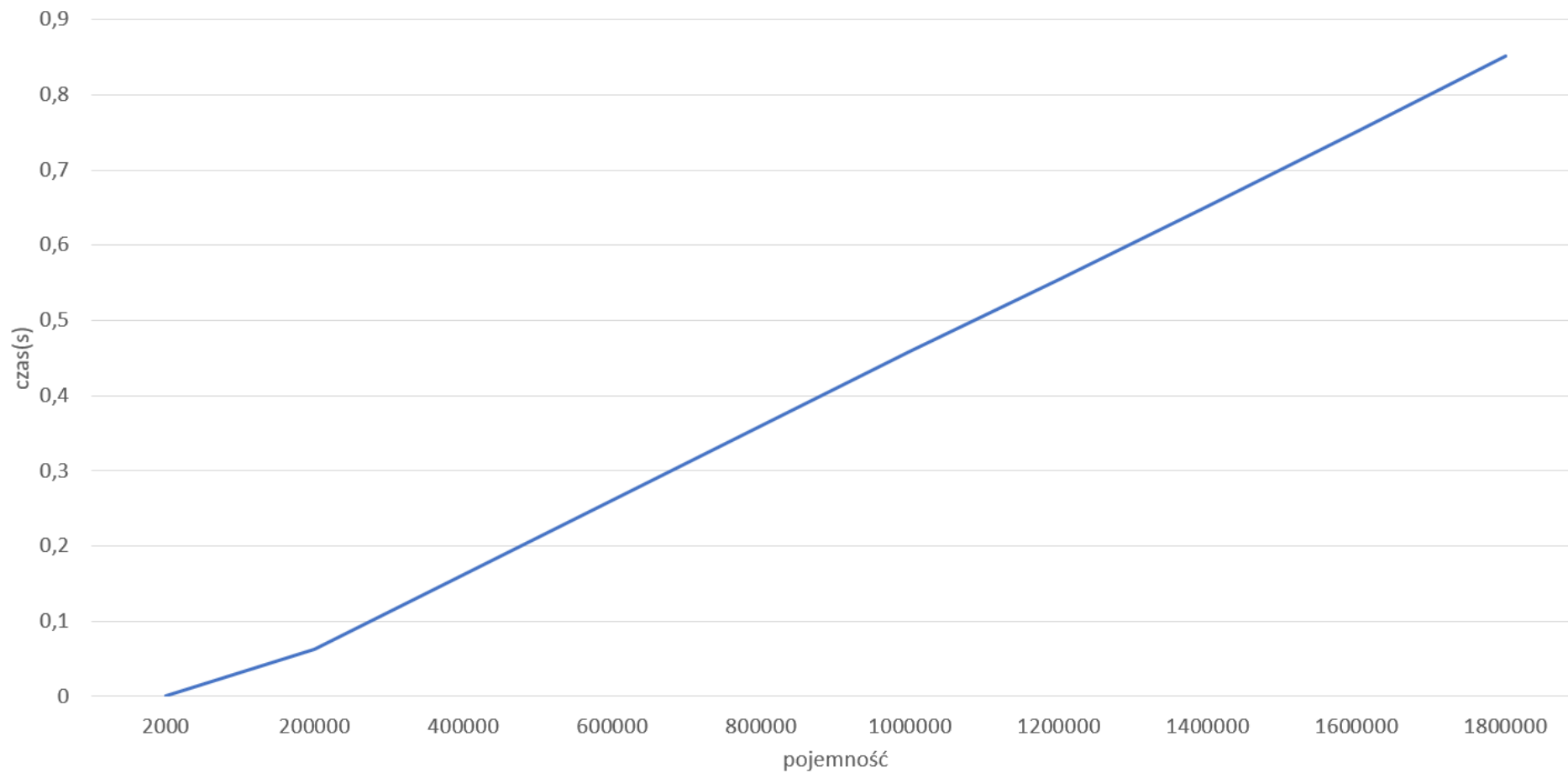
2.3) Czas wykonywania algorytmu siłowego w zależności od pojemności c i liczby elementów n .



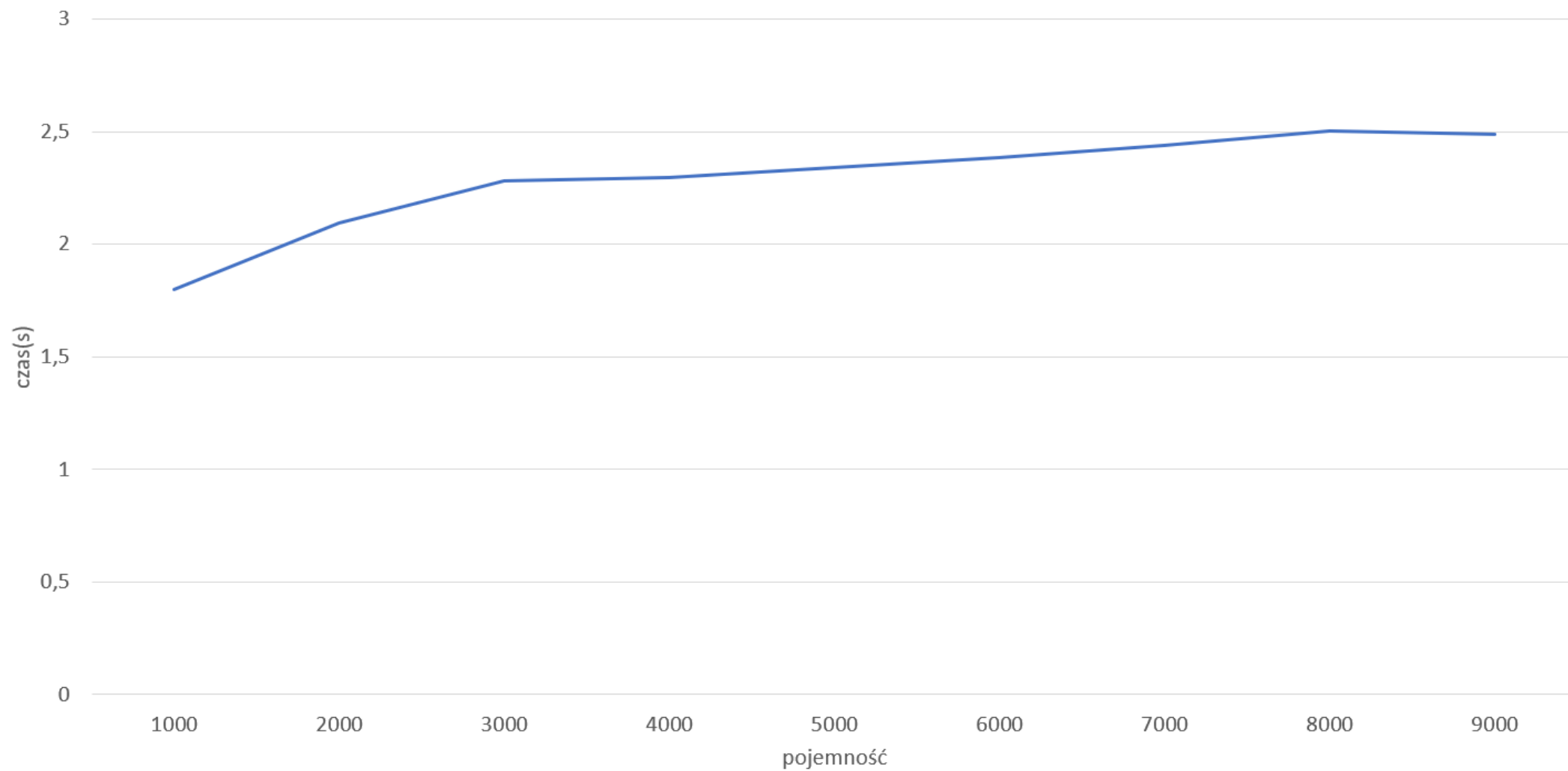
Powyższy algorytm ma złożoność wykładniczą, przez co ilość elementów w plecaku dla której zostanie on zastosowany jest mocno ograniczona, jeśli chcemy zakończyć obliczenia w sensownym czasie. Ze względu na to, że rozważa on zawsze każdą możliwą opcję, pojemność plecaka nie ma znaczącego wpływu na czas wykonywania.

3) Algorytm zachłanny.

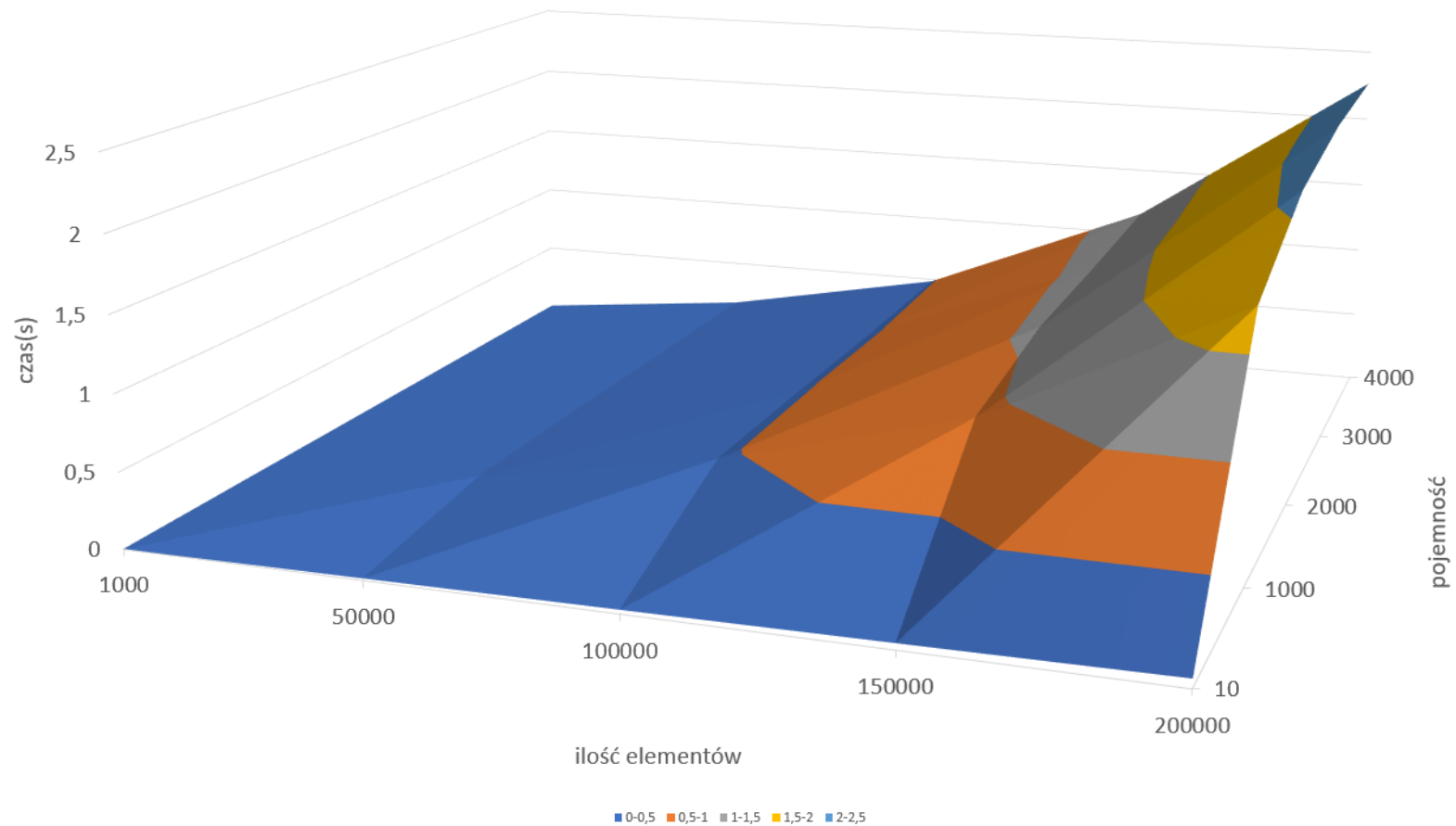
3.1) Czas wykonywania algorytmu zachłannego w zależności od liczby elementów (stała pojemność $c = 10$).



3.2) Czas wykonywania algorytmu zachłannego w zależności od pojemności (stała liczba elementów $n = 200000$).



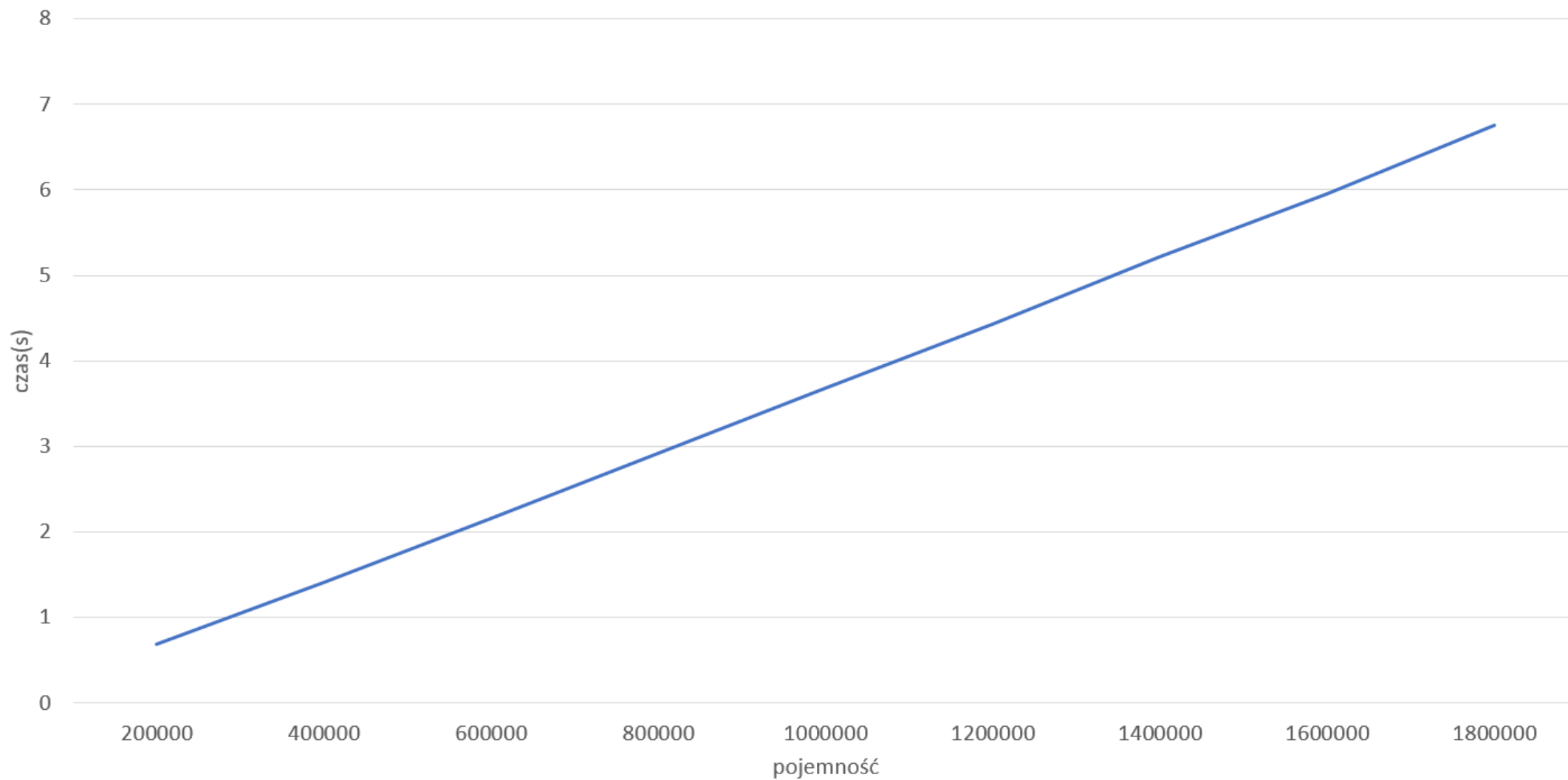
3.3) Czas wykonywania algorytmu zachłannego w zależności od pojemności c i liczby elementów n .



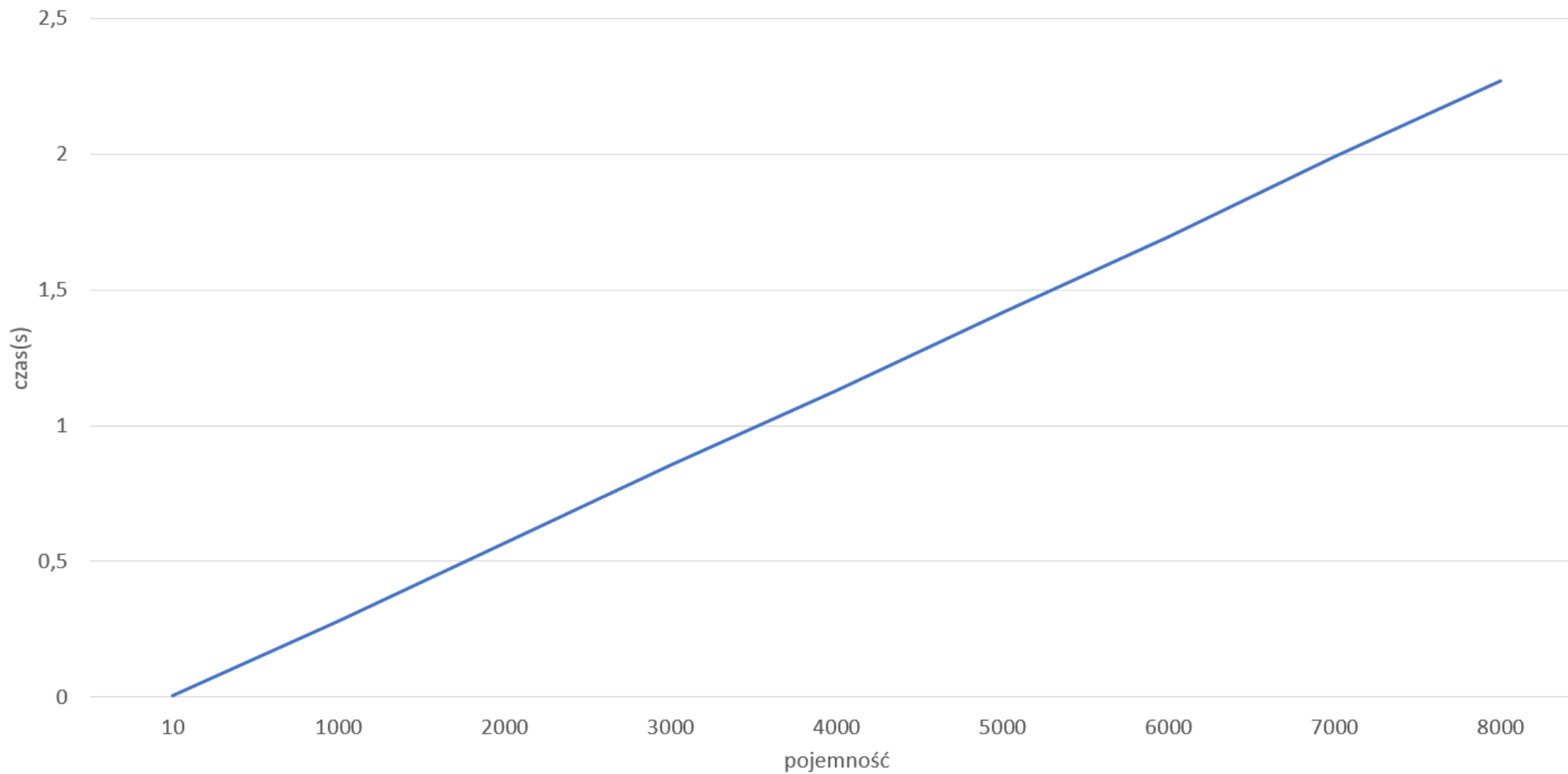
Złożoność algorytmu zachłannego wynosi $O(n \log n)$. Jest on zdecydowanie najszybszym z przedstawianych algorytmów, niestety nie daje on gwarancji optymalnego rozwiązania. Dodatkowo zauważyć można zależność czasu wykonywania nie tylko od liczby elementów ale także od pojemności plecaka.

4) Algorytm programowania dynamicznego.

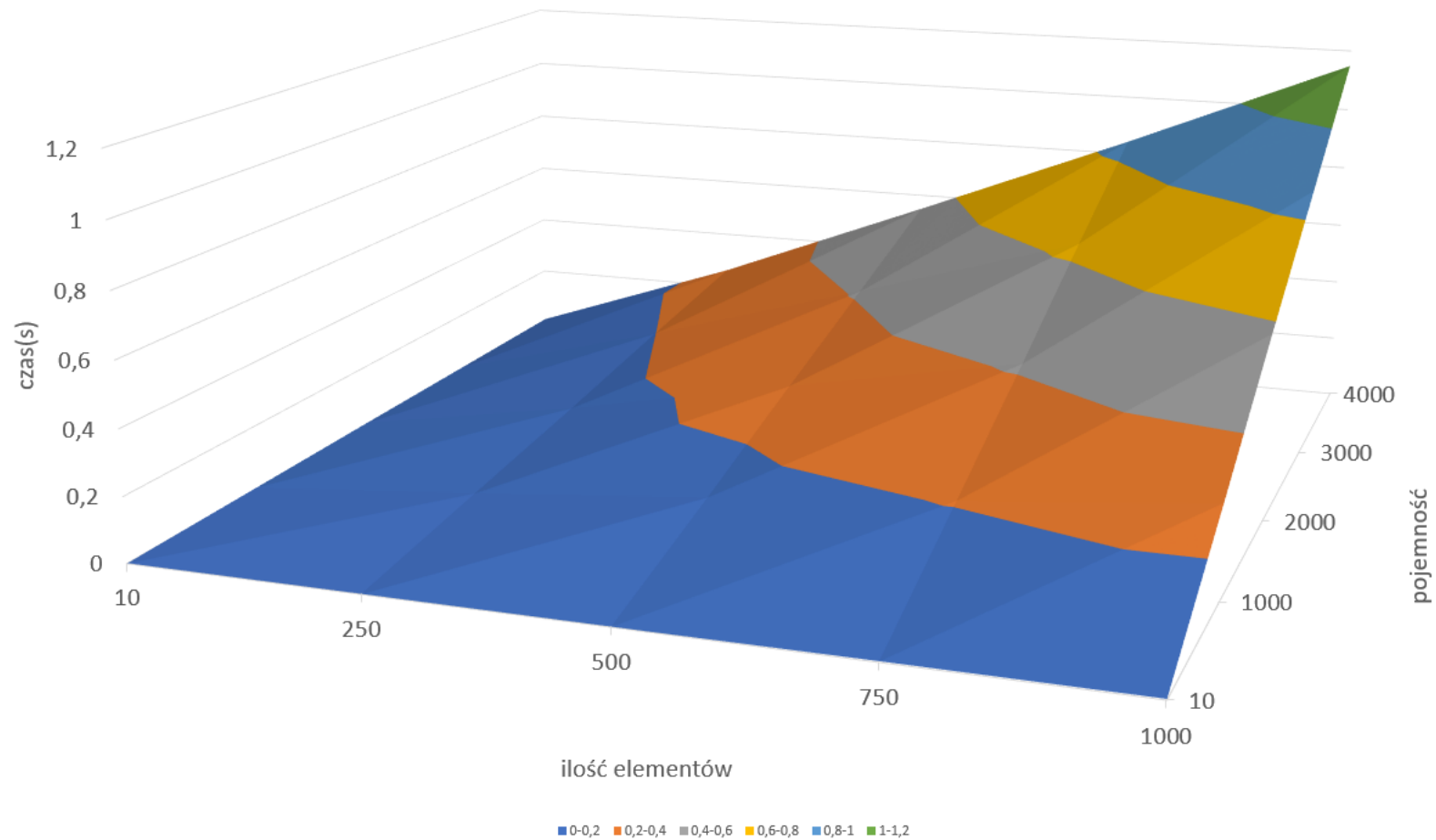
4.1) Czas wykonywania algorytmu dynamicznego w zależności od liczby elementów (stała pojemność $c = 10$).



4.2) Czas wykonywania algorytmu zachłannego w zależności od pojemności (stała liczba elementów $n = 1000$).

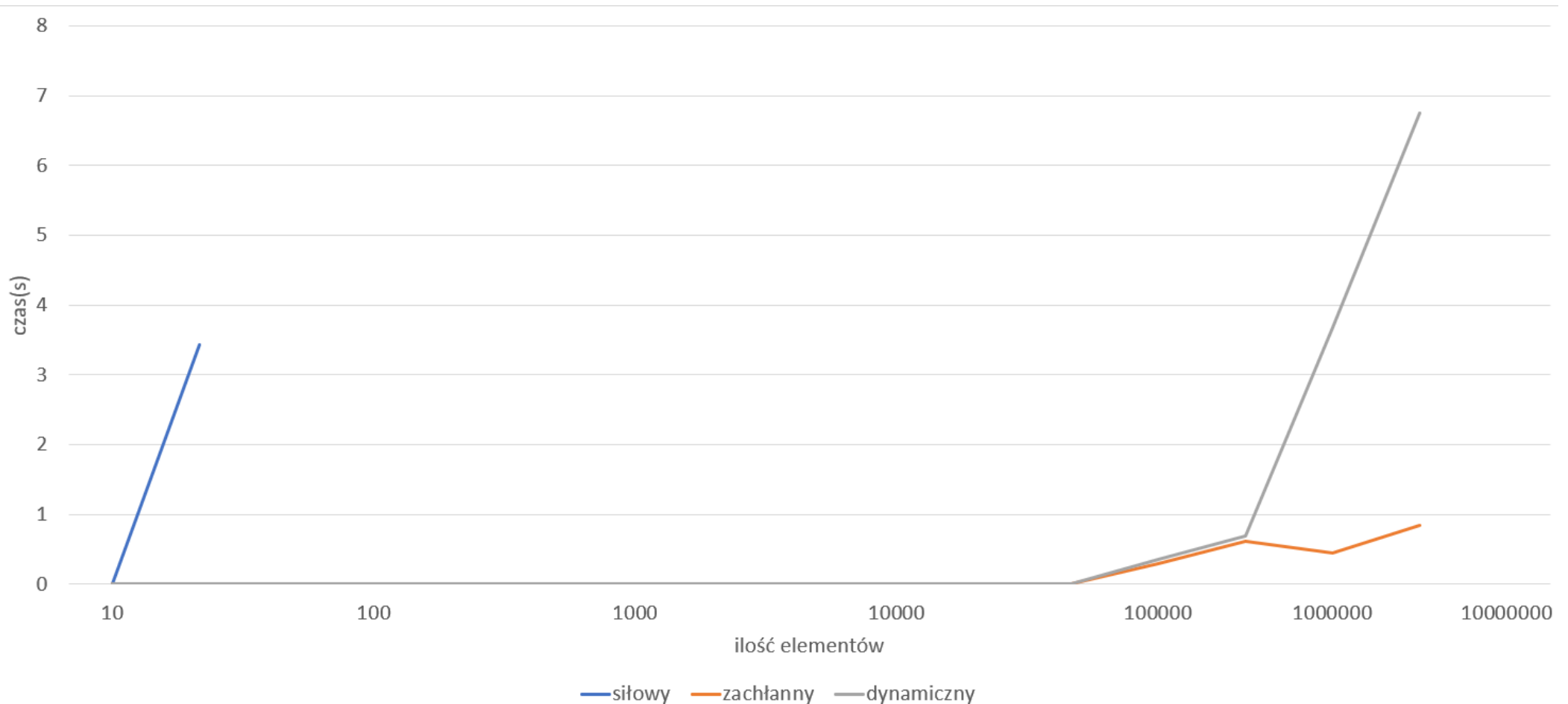


4.3) Czas wykonywania algorytmu zachłannego w zależności od pojemności c i liczby elementów n .



Złożoność algorytmu programowania Dynamicznego to $O(n * c)$, czyli pseudowielomianowa. Jego zaletą jest optymalność rozwiązań. Tak samo jak algorytm zachłanny, jego czas również zależy od ilości elementów jak i pojemności plecaka.

5) Wykres przedstawiający czas wykonywania wszystkich 3 algorytmów w zależności od liczby elementów(oś x jest w skali logarytmicznej).



6) Binarny problem plecakowy należy do klasy problemów NP-trudnych, jeśli mowa o problemie optymalizacyjnym, Problem decyzyjny za to należy do klasy problemów NP-zupełnych.

