

# Sprawozdanie - Problem Plecakowy

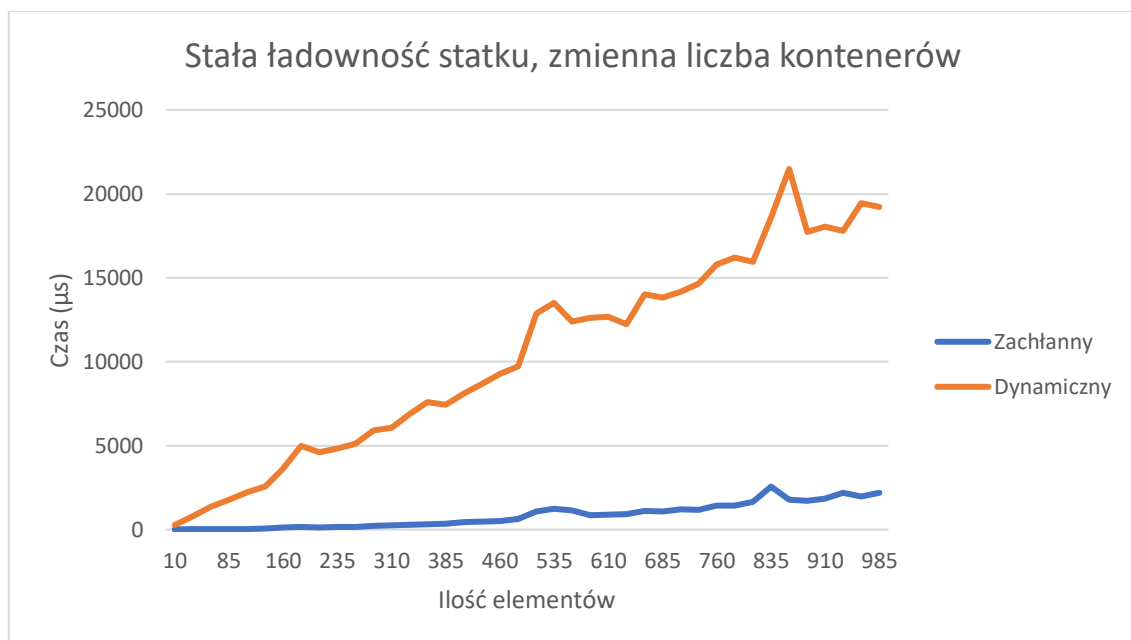
Szymon Szymankiewicz, Nikolas Szwargot

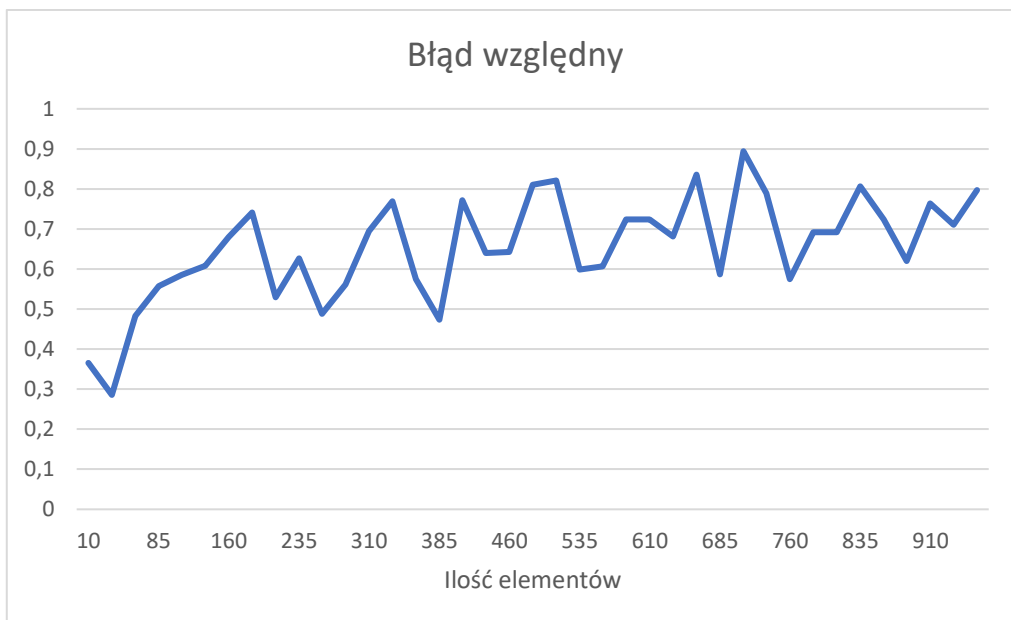
Lab 11, Informatyka, Semestr 2

## Wstęp

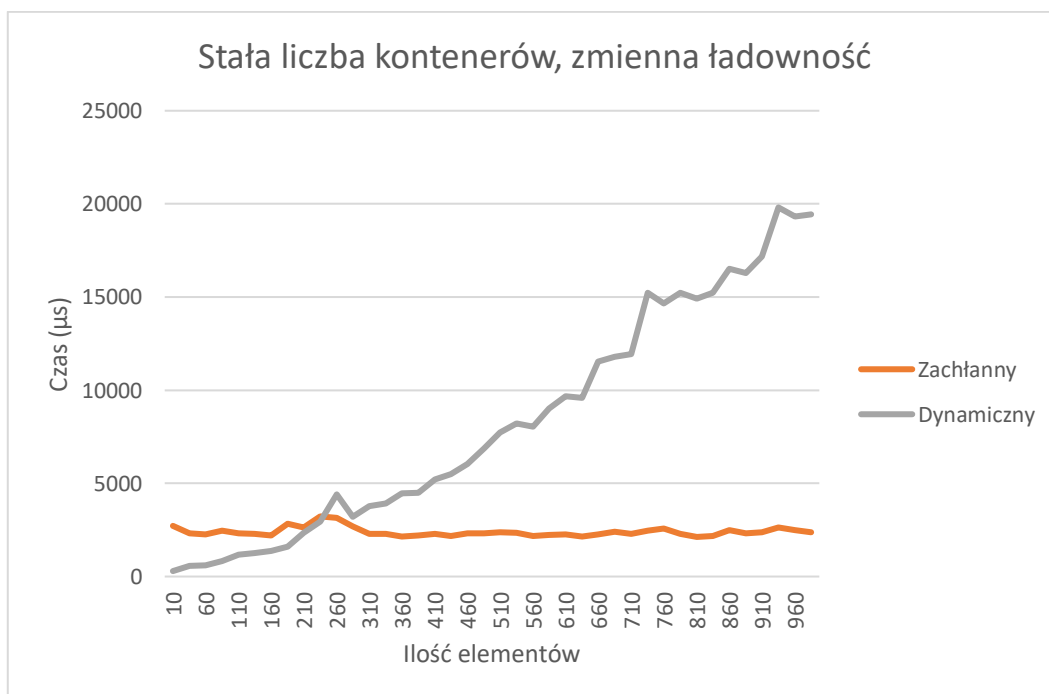
Celem naszego projektu było porównanie dwóch algorytmów i ich efektywności przy rozwiązywaniu problemu plecakowego. Są to algorytm zachłanny oraz algorytm programowania dynamicznego. Aby uzyskać lepsze wyniki na wykresach zostały przedstawione prędkości wykonywania obu algorytmów przy stałej ładowności, a następnie przy stałej liczbie kontenerów. W algorytmie zachłannym zostało użyte sortowanie bąbelkowe, które układa elementy nierosnąco względem ich wartości. Wszystkie programy zostały napisane w języku C++.

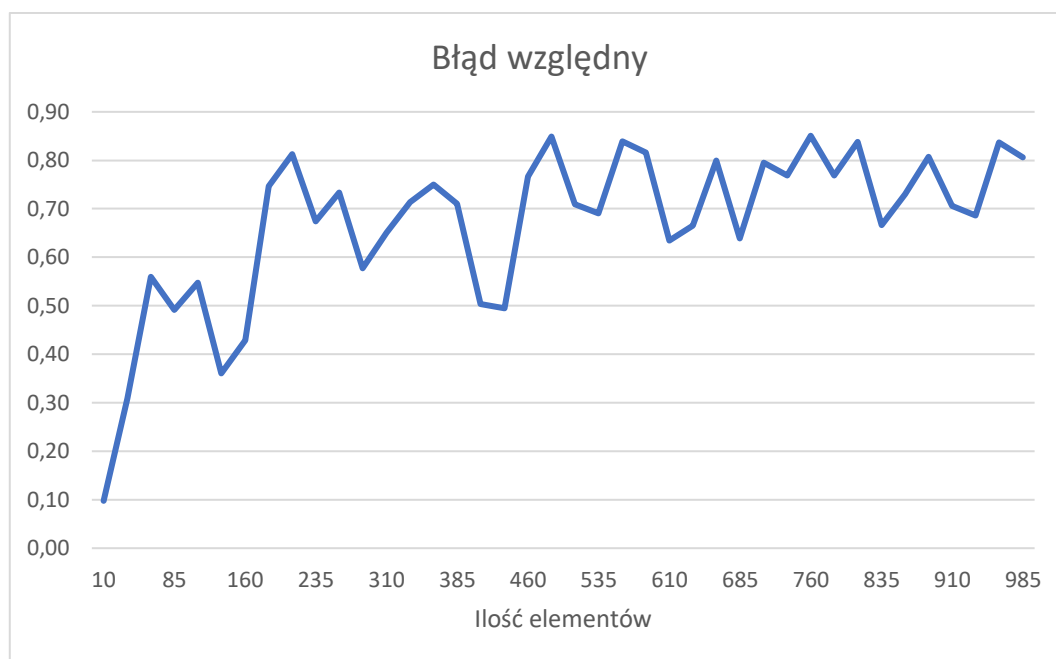
## Stała ładowność statku i zmienna liczba kontenerów





## Zmienna ładowność statku i stała liczba kontenerów





## Wnioski

### Algorytm programowania dynamicznego

Algorytm programowania dynamicznego w przeciwieństwie do algorytmu zachłannego daje nam rozwiązanie optymalne, czyli takie, gdzie wartość kontenerów, które są w stanie się zmieścić razem jest największa. Algorytm do rozwiązania problemu wykorzystuje macierz tworzoną poprzez rozwiązywanie mniejszych podproblemów. Wynika z tego, że jesteśmy w stanie z niej również odczytać rozwiązanie problemu plecakowego dla mniejszych problemów, niż nasz pierwotny. Jesteśmy w stanie również w łatwy sposób zwiększyć daną macierz. Poprzez istnienie samej macierzy potrzebna złożoność pamięciowa wynosi  $O(n * c)$ , gdzie  $n$  jest liczbą elementów z zbiorze, a  $c$  – pojemnością. Algorytm programowania dynamicznego zajmuje jednak więcej czasu na swoje wykonanie. Jest to spowodowane jego złożonością obliczeniową, która wynosi  $O(n * c)$ , gdzie  $n$  jest również liczbą elementów z zbiorze, a  $c$  – pojemnością. Jest to spowodowane tym, że algorytm podczas swojego działania musi przeszukać całą macierz o wymiarach  $c$  oraz  $n$ . Przez swoją złożoność (złożoność pseudowielomianowa) i prędkość wykonywania można wywnioskować, że programowanie dynamiczne służy do rozwiązywania problemów o klasie NP-zupełnych. Na podstawie pomiarów można stwierdzić, że rozmiar plecaka jak i liczba elementów podobnie wpływają na długość wykonywania się algorytmu. Wynika to z tego, że w przypadku pierwszym kolumny zwiększają się o tyle samo co wiersze w drugim przypadku, przez co całkowity rozmiar macierzy w obu przypadkach zostaje taki sam. Algorytm programowania dynamicznego może zostać użyty np. do znajdowania najdłuższego wspólnego podciągu, dopasowania ciągu znaków czy znajdowania optymalnego nawiasowania macierzy.

## Algorytm zachłanny

Jest to algorytm, który nie zawsze odnajdzie rozwiązanie optymalne. Algorytm zachłanny dokonuje wyboru rozwiązania na podstawie oceny sąsiedztwa i tego od czego oraz w jakiej kolejności zaczyna przeszukiwanie. W przypadku naszego rozwiązania algorytm sortuje elementy nierosnąco względem ich wartości. Następnie dobiera elementy, aż nie przekroczy limitu ładowności statku. Rozwiązania, które daje są rozwiązaniami suboptymalnymi. Mimo, iż nie mamy pewności, czy algorytm ten da nam najlepsze możliwe rozwiązanie ma on przewagę w szybkości wykonywania, gdyż wykonuje się znacznie szybciej niż algorytm programowania dynamicznego. Na prędkość wykonania nie wpływa znacząco ani zmieniająca się ilość elementów, ani maksymalna pojemność. Jego prędkość jest spowodowana złożonością obliczeniową, która wynosi  $O(n * n^2)$ . Składa się na nią złożoność potrzebna do wykonania sortowania ( $O(n^2)$ ), w tym przypadku zostało wykorzystane sortowanie bąbelkowe, a następnie w kolejnym kroku program buduje rozwiązanie poprzez przejście przez dane elementy ( $O(n)$ ). Jest to złożoność wielomianowa, więc algorytm zachłanny należy do klasy problemów P, co również pokazuje przewagę w prędkości wykonania nad algorytmem programowania dynamicznego. Jak można zauważyć w obu przypadkach błąd względny między algorytmem programowania dynamicznego, a zachłannym utrzymuje się na stosunkowo dużym poziomie (średnio między 0,6-0,8) co pokazuje, że niezależnie od metody tworzenia problemu plecakowego jakość rozwiązań generowanych przez algorytm zachłanny jest znacząco gorsza niż przy algorytmie programowania dynamicznego. Oznacza to, że w większości przypadków zamiast wyniku optymalnego został podany wynik suboptymalny. Algorytm zachłanny może mieć swoje zastosowanie np. jako algorytm do wyznaczania minimalnego drzewa rozpinającego, szeregowania zadań i wszędzie tam, gdzie ważniejszy jest czas niż dokładny wynik.