

# Porównanie prototypów

	Prototyp 1	Prototyp 2
Zastosowany algorytm	Breadth First Search ( przeszukiwanie wszerz )	Circle Packing
Zastosowane narzędzia	Java, Swing	Python, PyGame
Zasada działania funkcji oceny	$(wygoda - koszt) * bfs / dlugosc$  <div> Miejsce      siedzące      stojące  Wygoda      7                      3  Koszt        3                      1 </div>	$komfort + koszt + pojemno$  <ul style="list-style-type: none"> <li>• komfort ( liczba siedzeń )</li> <li>• koszt ( przesteń zajmowana przez siedzeń i wolna przestrzeń )</li> <li>• pojemność ( całkowita liczba pasażerów w autobusie )</li> <li>• wagi ( Ffactor-comfort, Ffactor-cost, Ffactor-capacity)</li> <li>• ceny ( Fseat-cost, Fspace-cost )</li> <li>• wartości minimalne i maksymalne</li> </ul>
Kodowanie genotypu	<ul style="list-style-type: none"> <li>• szerokość w liczbie mieszczących się w kolumnie ludzi</li> <li>• długość w liczbie mieszczących się w rzędzie ludzi</li> <li>• rozmieszczenie siedzeń i miejsc stojących</li> </ul>	<ul style="list-style-type: none"> <li>• długość</li> <li>• szerokość</li> <li>• maksymalna liczba miejsc siedzących</li> <li>• liczba rzędów</li> <li>• odległość między rzędami</li> <li>• parametr określający co ile siedzeń w rzędzie występuje separator</li> </ul>

	Prototyp 1	Prototyp 2
Selekcja osobników	<ul style="list-style-type: none"> <li>• metoda rankingowa</li> <li>• elitizm</li> </ul>	<ul style="list-style-type: none"> <li>• ruletka</li> <li>• elitizm</li> </ul>
Krzyżowanie osobników	<ul style="list-style-type: none"> <li>• Losowanie osobnika w którym zapisana jest pozycja siedzenia.</li> <li>• Podział autobusów na 3 kolumny i wymieszanie z drugim osobnikiem.</li> </ul>	<ul style="list-style-type: none"> <li>• Łączenie genotypów na “suwak”</li> </ul>
Mutacja	Jedno z miejsc uległo transformacji (osobniki wg. prawdopodobieństwa)	na losowej liczbie pozycji zostaje przypisana losowa wartość (osobniki wg. prawdopodobieństwa)
Szybkość działania	Wydajny obliczeniowo	Potrzeba dość dużej mocy obliczeniowej podczas symulowania większej liczby pasażerów.
Wyniki	Zastosowanie BFS umożliwia programowi ustalenie ścieżek dostępu do wszystkich siedzeń nawet przy skomplikowanym układzie autobusu.	Dzięki zastosowaniu algorytmu Circle Packing możliwe staje się dość dokładne ustalenie liczby osób która jest w stanie zmieścić się w autobusie.