

## Zespół nr: 1

Skład zespołu wraz z funkcjami:

1. **Paweł Szczepankiewicz:** koordynator, programista, autor dokumentacji
2. **Kamil Nalewajski:** programista, tester, strona graficzna
3. **Konrad Zdziarski:** programista, tester, strona graficzna

L.p.	Specyfikacja problemu (dane i wyniki)	Do jakich treści w zadaniu odnosi się algorytm	Zastosowane struktury danych	Informacje o zastosowanym algorytmie
1	<p><b>Dane:</b></p> <ul style="list-style-type: none"><li>• Średnia ilość jęczmienia na polu</li><li>• Produkcja w browarach</li><li>• Przepustowości dróg</li><li>• Ilość w karczmach</li></ul> <p><b>Wynik:</b></p> <p>→ reprezentacja danych w postaci struktur</p>	<p>“Zaproponuj jak informacje o polach, browarach, karczmach i drogach reprezentować w komputerze“</p>	<ul style="list-style-type: none"><li>▪ Lista</li><li>▪ Macierz sąsiedztwa</li><li>▪ Kolejka</li><li>▪ Stos</li></ul>	<ul style="list-style-type: none"><li>○ Brak zastosowanych algorytmów, użyto struktur.</li></ul>

2	<p><b>Dane:</b></p> <ul style="list-style-type: none"> <li>• Średnia ilość jęczmienia na polu</li> <li>• Produkcja w browarach</li> <li>• Przepustowości dróg</li> <li>• Ilość w karczmach</li> </ul> <p><b>Wynik:</b></p> <ul style="list-style-type: none"> <li>→ Maksimum piwa przewożonego do karczm</li> <li>→ Maksymalny przepływ w sieci</li> </ul>	<p>“Opracuj sposób znalezienia maksymalnej ilości piwa, która można dostarczyć do karczm w Shire”</p>	<ul style="list-style-type: none"> <li>▪ Graf skierowany z wagami</li> </ul>	<ul style="list-style-type: none"> <li>○ Algorytm <b>Edmondsa-Karpa</b> [maksymalny przepływ <math>O(V \cdot E^2)</math>]</li> <li>○ DFS/BFS</li> </ul>
3	<p><b>Dane:</b></p> <ul style="list-style-type: none"> <li>• Koszty odbudowy dróg</li> </ul> <p><b>Wynik:</b></p> <ul style="list-style-type: none"> <li>→ Optymalna droga bez strat zasobów</li> <li>→ Przepływ o najmniejszym koszcie</li> <li>→ Zachowaniu maksymalnego przepływu</li> </ul>	<p>“Zmodyfikujcie swoje rozwiązanie tak, żeby przy zachowaniu ilości przewożonego towaru, koszt naprawy dróg, po których poruszają się transporty był możliwie najmniejszy”</p>	<ul style="list-style-type: none"> <li>▪ Graf skierowany z wagami oraz kosztami</li> </ul>	<ul style="list-style-type: none"> <li>○ Algorytm Successive Shortest Path SSP [<math>O(F \cdot (V + E \cdot \log V))</math>]</li> <li>○ Algorytm <b>Dijkstry</b> z potencjałami [<math>O(E \cdot \log V)</math> na iterację]</li> </ul>

4	<p><b>Dane:</b></p> <ul style="list-style-type: none"> <li>• Ilość jęczmienia produkowanego na danej ćwiartce,</li> <li>• współrzędne ćwiartek</li> </ul> <p><b>Wynik:</b></p> <ul style="list-style-type: none"> <li>→ Całkowita ilość produkowanego jęczmienia</li> <li>→ Maksymalny przepływ</li> </ul>	<p>“Samwise kazał zebrać współrzędne punktów granicznych każdej ćwiartki (każda z ćwiartek okazała się być wielokątem wypukłym, rozłącznym z pozostałymi ćwiartkami). Wie też ile jęczmienia wyrasta na polu w poszczególnych ćwiartkach. Zmodyfikujcie swoje rozwiązanie uwzględniając te nowe informacje zebrane przez burmistrza Sama”</p>	<ul style="list-style-type: none"> <li>▪ Stosy</li> <li>▪ Wypukła otoczka</li> <li>▪ Sortowanie harmoniczne</li> </ul>	<ul style="list-style-type: none"> <li>○ Algorytm <b>Grahama</b> [wypukła otoczka <math>O(n \cdot \log n)</math>]</li> </ul>
---	--	---	--	--

5	<p><b>Dane:</b></p> <ul style="list-style-type: none"> <li>• Plik tekstowy</li> <li>• poszukiwane słowa</li> </ul> <p><b>Wynik:</b></p> <p>→ Pozycje poszukiwanych słów w tekście</p>	<p>“Burmistrz Marzy o tym, żeby szybko wyszukiwać w tych rozwiązaniach słów: „piwo”, „jęczmień”, „browar” oraz innych, które przyjdą mu kiedyś do głowy. Chciałby przetestować kilka sposobów wyszukiwania słów. Zaproponujcie odpowiednie rozwiązania”</p>	<ul style="list-style-type: none"> <li>▪ Lista</li> <li>▪ Słownik</li> <li>▪ Stos</li> </ul>	<ul style="list-style-type: none"> <li>○ Algorytm Naiwny [<math>O(n \cdot m)</math>]</li> <li>○ Algorytm <b>KMP Knutha-Morrisa-Pratta</b> [<math>O(n + m)</math>]</li> <li>○ Algorytm <b>Rabina-Karpa</b> [średnio <math>O(n + m)</math>]</li> <li>○ Algorytm <b>Trie</b> [<math>O(n + k)</math>, gdzie <math>k</math> to liczba dopasowań]</li> </ul>
6	<p><b>Dane:</b></p> <ul style="list-style-type: none"> <li>• Plik tekstowy</li> </ul> <p><b>Wynik:</b></p> <p>→ Skompresowany plik</p>	<p>“Komputer ma ograniczone zasoby”</p>	<ul style="list-style-type: none"> <li>▪ Drzewo binarne</li> <li>▪ Kolejka priorytetowa</li> </ul>	<ul style="list-style-type: none"> <li>○ Algorytm <b>Huffmana</b> [<math>O(n \cdot \log n)</math>, <math>n</math> – liczba symboli]</li> </ul>

7	<p><b>Dane:</b></p> <ul style="list-style-type: none"> <li>• Wymagania do użycia danego algorytmu</li> </ul> <p><b>Wynik:</b></p> <p>→ Reprezentacja graficzna projektu przy użyciu bibliotek</p>	Wizualizacja projektu	<ul style="list-style-type: none"> <li>▪ Obiekty</li> <li>▪ biblioteki graficzne</li> </ul>	<ul style="list-style-type: none"> <li>○ Brak algorytmu – użycie bibliotek do wizualizacji</li> </ul>
---	---	-----------------------	---	---