

## **Bioinformatyka II stopień, Pracownia informatyczna**

### **1 Zasady tworzenia aplikacji**

1. Określenie wymagań.
2. Specyfikacja – umowa pomiędzy zleceniodawcą a wytwórcą oprogramowania. Zleceniodawcy określają swoje potrzeby, a wytwórca doprecyzowuje je w taki sposób, aby umożliwić jednoznaczną realizację funkcjonalności produktu.  
Specyfikacja opisana jest językiem naturalnym – 21-28 października 2021 (sprawozdanie).
3. Projektowanie – analiza wymagań użytkownika na podstawie której zespół projektowy przystępuje do tworzenia modelu logicznego aplikacji – dane i wyniki. Podział na moduły – budowa schematu blokowego aplikacji – 11 listopada 2021 (sprawozdanie).
4. Projekt interface'u – 18 listopada 2021.
5. Opis algorytmów rozwiązujące poszczególne moduły – 25 listopada 2021 (sprawozdanie).
6. Projektowanie – przygotowanie danych testowych sprawdzających poprawność działania poszczególnych modułów – 9 grudnia 2021.
7. Projektowanie – model logiczny poszczególnych modułów oprogramowania (schemat blokowy modułu), określenie wyjątków – 13 stycznia 2022 (sprawozdanie na koniec semestru).
8. Kodowanie modułów, łączenie modułów w spójny program – marzec – kwiecień 2022.
9. Testowanie modułów i całej aplikacji – koniec kwietnia 2022.
10. Opis poszczególnych modułów z wyszczególnieniem użytych zmiennych, ich znaczenia lokalnego i globalnego, opis procedur itp - połowa maja 2022.
11. Program instalacyjny – koniec maja 2022.
12. Help programu – koniec maja 2022.

## 2 Zadania

Zgodnie z powyższymi krokami należy w języku python w wersji 3.8 lub wyższej napisać użytkową aplikację wykonującą jedno z poniższych zadań.

1. Baza GBIF - the Global Biodiversity Information Facility. Na zadanym obszarze należy wyznaczyć listę występujących gatunków oraz ich rozłożenie – prezentacja tekstowa i graficzna (mapa).  
Na podstawie pracy Paula Flemons'a A web-based GIS tool for exploring the world's biodiversity: The Global Biodiversity Information Facility Mapping and Analysis Portal Application (GBIF-MAPA) (załącznik – Flemons.pdf).

2. Baza iNaturalist - występowanie gatunków zagrożonych na danym obszarze.
3. Rozprzestrzenianie się epidemii COVID-19 – zmienność w czasie i obszarze, porównanie ze stopniem zaszczepienia populacji.
4. Data Cleaning – podręcznik (rozdziały 2-5) (załącznik Data Cleaning.pdf).
5. Algorytmy wyszukiwania wzorca w tekście – zastosowanie do wyszukiwania wzorca białek. Patrz:

<https://docplayer.pl/47383200-Temat-algorytmy-wyszukiwania-wzorca-w-tekscie.html>

Zaprogramować algorytmy Knutha – Morrisa – Pratta (KMP) oraz Rabina-Karpa (RK).

6. Struktury drzewiaste w opisie taksonomii świata roślin (zwierząt), budowa struktury, jej modyfikacja, prezentacja.
7. Wizualizacja dopasowania co najmniej trzech długich sekwencji – rzędu tysięcy aminokwasów.
8. Używając pythonowskiej biblioteki bio napisz program porównujący wyniki wielo-sekwencyjnego dopasowania sekwencji aminokwasów. Wykorzystaj pakiety clustalW oraz T\_coffee.
9. Testy rangowe (co najmniej 5 różnych testów) – program nie może korzystać z bibliotek statystycznych (załącznik MBPzS\_wykład\_10.pdf).
10. Łączenie w skupienia – metodami analizy skupień, interpretacja graficzna wyników. Danymi są ciągi obserwacji obiektów. Każdy obiekt może być opisany przez kilka cech. Program ma zawierać analizę obserwacji odstających.
11. W pracy pt: „The diagonal-traverse homology search algorithm for locating similarities between two sequences” (załącznik nar0344-0324.pdf) przedstawiono algorytm porównania dwóch sekwencji DNA. Napisz program kodujący powyższy algorytm. Wyniki przedstaw graficznie i tekstowo. Program ma badać sekwencje aminokwasowe (nukleotydowe) o długości rzędu kilku tysięcy aminokwasów (nukleotydów).
12. Algorytm Needlemana-Wunscha – dopasowanie lokalne (algorytm Smitha–Watermana). Dla różnych macierzy punktacji napisz program pokazujący różnice w dopasowaniach lokalnych.
13. Na stronie

<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3638164/>

prezentowany jest algorytm FOGSAA: „Fast Optimal Global Sequence Alignment Algorithm”. Napisz program go kodujący. Otrzymane wyniki porównaj z algorytmem Needelmana-Wunscha.

14. Lokalna (postawiona na localhost) baza danych dowolnej hodowli – jest budowa i obsługa. Baza ma mieć strukturę relacyjną.
15. Lokalna (postawiona na localhost) baza danych ogrodnictwa – jest budowa i obsługa. Baza ma mieć strukturę relacyjną.