# Bioinformatyka - ćwiczenie 6

## Wykorzystanie Pythona w bioinformatyce - część II

Czas na oddanie finalnego raportu: 14 dni

[Zasady obowiązujące na przedmiocie](https://docs.google.com/document/d/1WQFgVABO-VEk_DY7aKIpzlQuQimv_4fi06A13ZXQRUU/edit?usp=sharing), [plik z ocenami](https://docs.google.com/spreadsheets/d/1u1qCM90qtG6h3g_3T10HMl1p-lb2I1W9A-B-g1zeTwU/edit?usp=sharing)

Pliki wynikowe należy przesłać poprzez platformę MS Teams. Aby zaliczyć ćwiczenie należy przesłać rozwiązania trzech zadań. Przesłanie rozwiązania dwóch zadań - ocena w dół. Przesłanie rozwiązania jednego zadania - dwie oceny w dół.

Cel: W ramach tego ćwiczenia studenci zapoznają się z zaawansowanymi metodami analizy sekwencji biologicznych, takimi jak porównywanie sekwencji DNA (alignment), projektowanie starterów oraz obliczanie ich parametrów dla reakcji PCR.

### **Teoria**

**Porównywanie sekwencji DNA**, znane jako **alignment**, polega na dopasowywaniu dwóch lub więcej sekwencji w celu znalezienia obszarów homologii. Homologia ta może wskazywać na wspólne pochodzenie ewolucyjne lub funkcjonalne powiązania między sekwencjami. Istnieją dwa główne typy przyrównania:

* **Globalne przyrównanie (Needleman-Wunsch)** – dopasowanie na całej długości sekwencji, przydatne do porównywania sekwencji o podobnej długości.
* **Lokalne przyrównanie (Smith-Waterman)** – dopasowanie najlepiej pasujących fragmentów sekwencji, wykorzystywane do porównywania częściowych podobieństw.

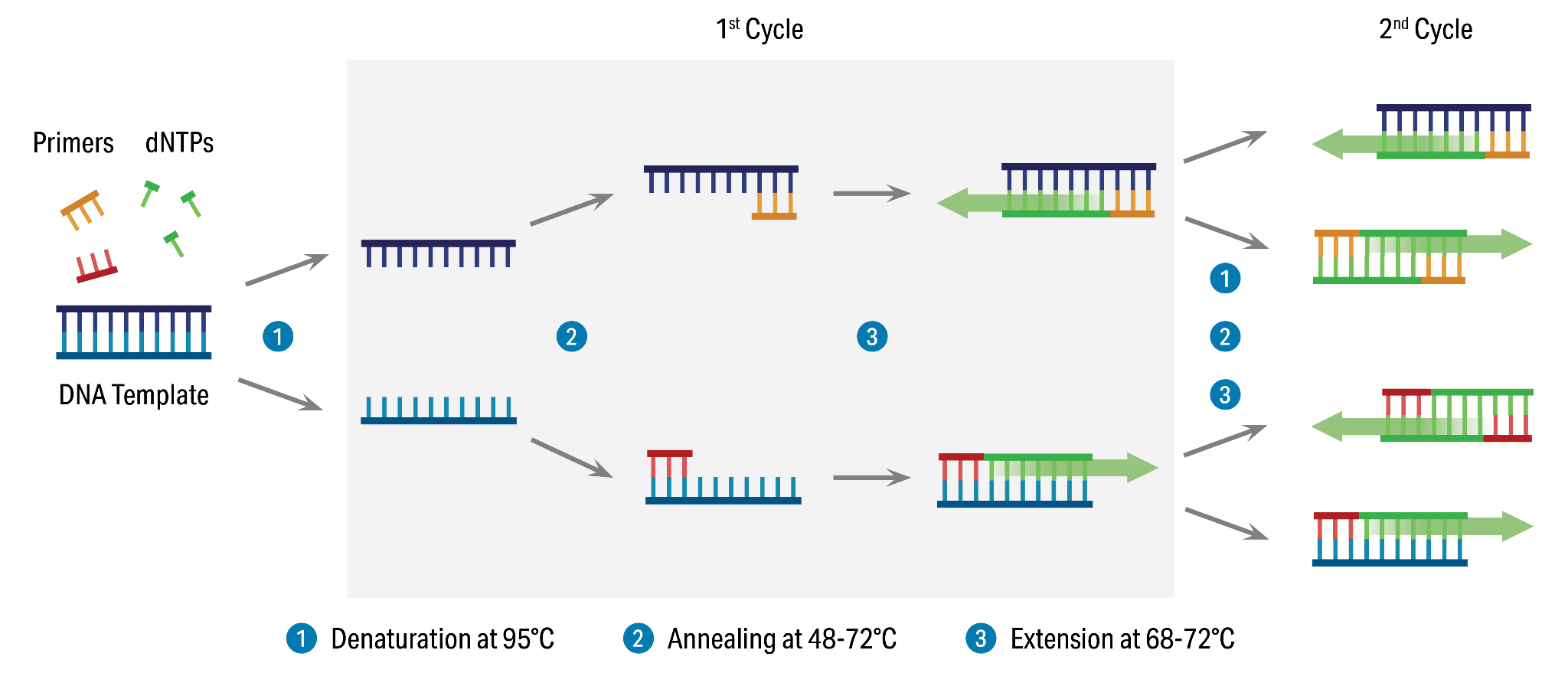
Podczas przyrównania ważne są następujące parametry:

* **Procent identyczności** – określa liczbę identycznych nukleotydów w dopasowanych pozycjach.
* **Luki** – wstawienia lub delecje w jednej z sekwencji.
* **Wartości scoringowe** – wynik obliczany na podstawie algorytmu i macierzy scoringowej.

Moduł **Align** w Biopython umożliwia przeprowadzanie przyrównania sekwencji przy użyciu wbudowanych algorytmów i macierzy scoringowych. Przykład lokalnego przyrównania:

| from Bio.Align import pairwise2 from Bio.Seq import Seq  seq1 = Seq("ACTG") seq2 = Seq("ACGG") alignments = pairwise2.align.localxx(seq1, seq2)  for alignment in alignments:  print(alignment) |
| --- |

**Reakcja łańcuchowa polimerazy (PCR)** jest techniką amplifikacji (powielania) fragmentów DNA, fundamentalną w badaniach molekularnych i diagnostyce genetycznej. Ważnym elementem PCR są **primery (startery)** – krótkie odcinki DNA komplementarne do wybranych fragmentów sekwencji, które inicjują amplifikację.



Projektowanie starterów wymaga uwzględnienia następujących parametrów:

1. **Temperatura topnienia (Tm)** – określa stabilność wiązań wodorowych między primerem a matrycą DNA. Optymalny zakres Tm dla primerów to 50–60°C.
2. **Zawartość GC** – proporcja nukleotydów G i C w primerze. Wyższa zawartość GC stabilizuje wiązania.
3. **Unikanie struktur wtórnych** – primer nie powinien tworzyć dimerów ani struktur typu hairpin, które mogą zakłócać amplifikację.
4. **Specyficzność** – primer powinien wiązać się tylko z docelowym fragmentem sekwencji, aby uniknąć amplifikacji niespecyficznych regionów.

Biblioteka **Primer3-py** umożliwia automatyczne projektowanie primerów oraz obliczanie ich kluczowych parametrów. Dzięki tej bibliotece można:

* Wyznaczyć pary primerów (forward i reverse) dla określonej sekwencji.
* Określić optymalną długość startera, Tm i zawartość GC.
* Uwzględnić regiony flankujące lub wykluczyć określone obszary sekwencji.

Przykład użycia Primer3-py:

| from primer3 import calcTm   sequence = "ACTGCGTACGTAGCTAGCTAGC"  primer = "ACTGCGTACG"  tm = calcTm(primer)  print(f"Temperatura topnienia: {tm:.2f}°C") |
| --- |

### **Ogólne wytyczne do zadań praktycznych**

Zadania są oparte na problemach praktycznych. Twoim celem jest znalezienie rozwiązania przy użyciu języka Python i biblioteki Biopython, korzystając z dostępnych źródeł informacji. Sposób rozwiązania problemu jest dowolny, ale kod powinien być możliwie jak najprostszy i czytelny. Jeśli korzystasz z narzędzi sztucznej inteligencji, dodaj tę informację w swoim rozwiązaniu w formie komentarza. Pamiętaj, że prowadzący może poprosić Cię o wyjaśnienie wybranego fragmentu kodu, dlatego warto rozumieć każdy jego element.

### **Wymagania dotyczące oprogramowania**

**Python 3.X** z zainstalowaną biblioteką Biopython i primer3-py. Najprostszym sposobem skonfigurowania środowiska jest zainstalowanie Anacondy z [Anaconda Distribution Webpage](https://www.anaconda.com/download/success) i z jego pomocą zainstalowanie Biopython i primer3-py:

| conda install -c conda-forge biopython  conda install -c conda-forge primer3-py |
| --- |

### **Zadanie I: Biopython przyrównanie sekwencji**

1. Upewnij się, że Python jest zainstalowany.
2. Zapoznaj się z [dokumentacją Biopython](https://biopython.org/docs/latest/Tutorial/chapter_align.html) dla Sequence alignments.

**Analiza wszystkich możliwych przyrównań**

1. Użyj następujących sekwencji DNA:

seq\_1 = "AAAACCCCTTGGTTTT"

seq\_2 = "AAACACCCTTTGTTTA"

1. Utwórz obiekt aligner jako instancję Align.PairwiseAligner().
2. Wykorzystaj dostarczony plik **script\_1.py** jako punkt startowy i zaimplementuj następujące kroki:
   1. Oblicz wynik przyrównania (score):
      1. Wyznacz wartość dopasowania pomiędzy sekwencjami.
   2. Wyświetl przyrównania sekwencji:
      1. Wypisz kilka pierwszych przyrównań.
   3. Przeanalizuj macierze koordynat:
      1. Dla pierwszych trzech przyrównań wypisz ich macierze koordynat.
      2. Wyjaśnij, jakie informacje można odczytać z macierzy koordynat (np. fragmenty pasujących regionów) (zapisz jako komentarz w kodzie).
   4. Liczba przyrównań z niezerowym błędem niedopasowania (mismatch):
      1. Zaimplementuj funkcję liczącą przyrównania, które zawierają co najmniej jedno niedopasowanie.

**Znalezienie fragmentu z największą liczbą pasujących nukleotydów**

1. Wykorzystaj dostarczony plik **script\_2.py** i ujęte w nim sekwencje jako punkt startowy.
2. Znajdź najdłuższy fragment pasujących par nukleotydów spośród wszystkich możliwych przyrównania sekwencji. Przykład:

| target 0 ATTATAGC-AGTG-TA 14  0 ||||-|||-||-|-|| 16  query 0 ATTA-AGCCAG-GGTA 14  ˆˆˆˆ - fragment z największa ilością pasujących nukleotydów |
| --- |

W tym przypadku:

Fragment o największej liczbie pasujących nukleotydów zaczyna się na pozycji **0**.

Długość tego fragmentu wynosi **4**.

1. Za pomocą platformy MS Teams prześlij skrypty Pythona **login\_imie\_nazwisko\_script\_1.py i** **login\_imie\_nazwisko\_script\_2.py**. Upewnij się, że kod jest czytelny i odpowiednio skomentowany.

### **Zadanie II: Primer3 analiza parametrów primerów**

1. Zapoznaj się z dokumentacją [primer3-py](https://libnano.github.io/primer3-py/).
2. Do pracy użyj dostarczonego pliku **primers.txt**.
3. Dla każdego primera wczytanego z pliku:
   1. Oblicz temperaturę topnienia (Tm).
   2. Sprawdź możliwość utworzenia struktury homodimer i zapisz odpowiedź jako Tak/Nie (w zależności od wyniku analizy).
   3. Sprawdź możliwość utworzenia struktury hairpin i zapisz odpowiedź jako Tak/Nie.

Wzór formatu rozwiązania:

| Primer,Tm,Homodimer,Hairpin  TACGTCAATCCCAGAACGCT,56.98,Tak,Nie  GACAACATGGGCGAATTGGA,56.70,Tak,Nie  AGTTTCCTTCACCCGTCCTT,56.74,Tak,Nie  TAGTGATCACTGCCCACGTA,56.04,Tak,Tak  CCGTGTACTGGTTAAGCTCA,55.14,Tak,Nie  CCGCCTACGACATACGAAAA,55.97,Tak,Tak |
| --- |

1. Za pomocą platformy MS Teams:
   1. Prześlij plik **login\_imie\_nazwisko\_script\_3.py** zawierający kod Pythona.
   2. Dołącz plik wynikowy **primer\_analysis.csv** z danymi wyjściowymi.

### **Zadanie III: Primer3 projektowanie starterów do reakcji**

1. Zapoznaj się z dokumentacją [primer3-py](https://libnano.github.io/primer3-py/), w szczególności sekcję “Primer design”.
2. Użyj poniższej sekwencji DNA:

| GCTTGCATGCCTGCAGGTCGACTCTAGAGGATCCCCCTACATTTT  AGCATCAGTGAGTACAGCATGCTTACTGGAAGAGAGGGTCATGCA  ACAGATTAGGAGGTAAGTTTGCAAAGGCAGGCTAAGGAGGAGACG  CACTGAATGCCATGGTAAGAACTCTGGACATAAAAATATTGGAAG  TTGTTGAGCAAGTNAAAAAAATGTTTGGAAGTGTTACTTTAGCAA  TGGCAAGAATGATAGTATGGAATAGATTGGCAGAATGAAGGCAAA  ATGATTAGACATATTGCATTAAGGTAAAAAATGATAACTGAAGAA  TTATGTGCCACACTTATTAATAAGAAAGAATATGTGAACCTTGCA  GATGTTTCCCTCTAGTAG |
| --- |

1. Twoim zadaniem jest znalezienie przynajmniej jednej pary primerów służących do amplifikacji fragmentu powyższej sekwencji od 100 do 250 nukleotydu w reakcji PCR. Wynik podaj w następującej postaci:

| Left primer Right primer Start End  AAGGCAGGCTAAGGAGGAGA -- TGCCTTCATTCTGCCAATCT 80 290  GCTAAGGAGGAGACGCACTG -- TGCCTTCATTCTGCCAATCT 85 273 |
| --- |

gdzie Start oznacza początek amplifikowanego fragmentu, a End jego koniec.

1. Za pomocą platformy MS Teams:
   1. Prześlij plik **login\_imie\_nazwisko\_script\_4.py** zawierający kod Pythona.
   2. Dołącz plik wynikowy **PCR\_primers.txt**.