

## Esercitazione 2011-05-10: utilizzo delle principali classi della libreria BioJava

### Riferimenti:

- Codici messi a disposizione sul sito ufficiale di BioJava:  
<http://www.biojava.org/wiki/BioJava:CookBookLegacy> .
- Slides su BioJava del corso di Algoritmi e Linguaggi per Bioinformatica.
- Codici soluzione: i codici soluzione saranno rilasciati al termine della lezione sul sito del corso ([www.albertocastellini.tk](http://www.albertocastellini.tk)).

### Esercizi:

1. Creazione alfabeti dna, rna, protein, protein-term, stampa dei loro elementi/simboli e stampa del numero di elementi di ogni alfabeto.  
→ Codice soluzione: progetto BioJava1Test, classe AlphabetExample
2. Creazione di SymbolList per le seguenti stringhe (String):
  - a. Dna: atcggtcggctta
  - b. Rna: auugccuacauaggc
  - c. Proteine: AGFAVENDSAE stampa delle SymbolList.  
→ Codice soluzione: progetto BioJava1Test, classe StringToSymbolList
3. Generare una SymbolList per la sequenza di rna auggcaccguccagauu. Prendere il primo simbolo e stamparne il nome, prendere la sottolista dei primi tre simboli e stamparla, prendere la sottolista degli ultimi tre simboli e stamparla.  
→ Codice soluzione: progetto BioJava1Test, classe SubSequencing
4. Generare una SymbolList del dna atgccgaatcgtaa, convertirla in una SymbolList di rna (i.e., trasformare le T in U). Convertire la SymbolList di dna iniziale nella SymbolList di rna che si otterrebbe tramite il processo biologico di trascrizione (metodo DNATools.transcribeToRNA). Stampare le tre sequenze ottenute.  
→ Codice soluzione: progetto BioJava1Test, classe TranscribeDNAtoRNA
5. Generare una SymbolList del dna atgccgaatcgtaa, determinare e stampare la SymbolList della stringa inversa (reverse), quella della complementare (complement) e quella della inversa complementare (reverse complement).  
→ Codice soluzione: progetto BioJava1Test, classe ReverseComplement
6. Generare una SymbolList del dna atgccgaatcgtaa, convertirla in una SymbolList di rna (i.e., trasformare le T in U). Tradurla (translate) nella corrispondente sequenza (SymbolList) di amminoacidi. Stampare le sequenze ottenute.  
→ Codice soluzione: progetto BioJava1Test, classe Translate
7. Generare un alfabeto di codoni (triplette di nucleotidi) a partire dall'alfabeto del dna. Stampare tutti i simboli dell'alfabeto di codoni.  
→ Codice: progetto BioJava1Test, classe CrossProduct

8. Creare le seguenti viste (View) della SymbolList di dna atgcccgcgtaa: vista a singoli nucleotidi, vista a codoni (SymbolListViews.windowedSymbolList(dna, 3)), a triplette (SymbolListViews.orderNSymbolList(dna, 3)). Stampare le SymbolList ottenute.  
→ Codice: progetto BioJava1Test, classe CodonView
  
9. Creare di un simbolo ambiguo (BasisSymbol) di tipo cytosine, thymine (Y) e stampare i simboli atomici a cui si riferisce  
→ Codice: progetto BioJava1Test, classe Ambiguity