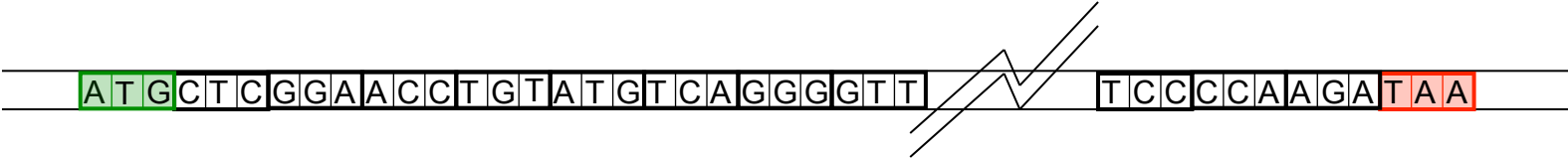


## 2. Genes and proteins

- The sequence as a model of DNA
- Genes: from Mendel to molecular biology
- The genetic code
- **A translation algorithm**
- Implementing the genetic code
- Algorithms + data structures = programs
- The algorithm design trade-off
- DNA sequencing
- Whole genome sequencing
- How to find genes?

# A translation algorithm

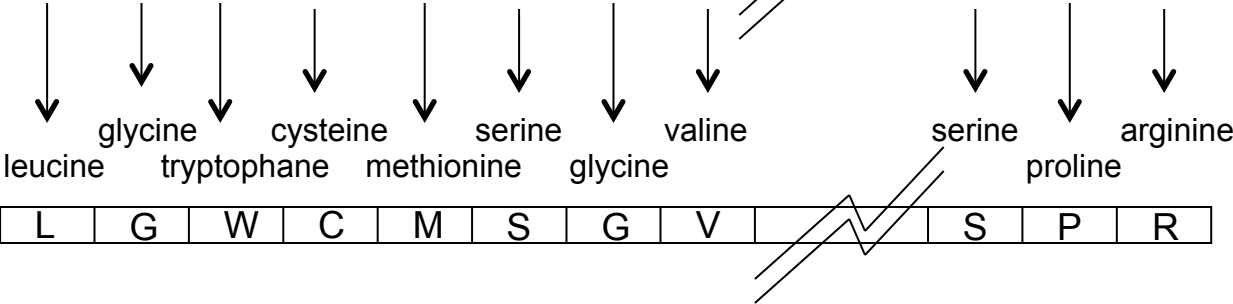
# Transcription and translation



Transcription



Translation



# A translation algorithm

- For every triplet of the coding DNA sequence
  - look up in the array implementing the genetic code and retrieve the corresponding amino acid or stop codon
  - add the AA one-letter name to the protein sequence

CTTATGGCATTGACTGATTCATGC\*

CTT ATG GCA TTG ACT GAT TCA TGC



L M A L T D S C \*

```
indexDNA, indexProtein: integer  
DNAsequence, ProteinSequence: character string [1:*]  
indexProtein, indexDNA ← 1  
repeat  
  ProteinSequence [indexProtein] ← lookupGeneticCode (DNAsequence  
    [indexDNA], DNAsequence [indexDNA+1], DNAsequence [indexDNA+2])  
  indexDNA ← indexDNA + 3  
  indexProtein ← indexProtein + 1  
until DNAsequence [indexDNA] = "*"   
ProteinSequence [indexProtein] ← "*" 
```

lookupGeneticCode is a function  
which is written separately to reduce the difficulty  
of writing the entire algorithm at once

# And if the number of letters is not a multiple of 3?

```
indexDNA, indexProtein: integer
```

```
DNAsequence, ProteinSequence: character string [1:*]
```

```
indexProt, indexDNA ← 1
```

```
repeat
```

```
  ProteinSequence [indexProt] ← lookupGeneticCode (DNAsequence  
  [indexDNA], DNAsequence [indexDNA+1], DNAsequence [indexDNA+2])
```

```
  indexDNA ← indexDNA+3
```

```
until DNAsequence [indexDNA] = "*" or
```

```
  DNAsequence [indexDNA+1] = "*" or
```

```
  DNAsequence [indexDNA+2] = "*"
```

```
ProteinSequence [indexProt] ← "*"
```