

ACCESS 2010
week 1 a.m.
Part 1 of your group project assignment

The data from *Cracking_the_Code.pdf* that you have been given can be roughly summarized as follows:

1. When a biologist gives a sequence of nucleotide bases (the encrypted message) from the left column of the table, out comes one of the corresponding strings of amino-acids from the right column (the decrypted message). For example, UCUCUC..., translates into ...ser-leu-ser-leu... (which is the same as ...leu-ser-leu-ser...). You also know Nierenberg's result.
2. Which of the possible amino-acid chains you get depends on where the ribosome starts reading the encrypted message.
3. You know (with the benefit of hindsight that the original scientists didn't have) that
 - several different nucleotide sequences can code the same amino-acid
 - some sequences of nucleotides mean "stop the translation"
 - there are 20 standard amino-acids

Additionally, we will assume that

4. The ribosome reads systematically, so that it always begins reading a new codon the same number of nucleotide bases from where it began reading the previous one. We'll call this number the *reading restart interval*.
5. All codons have the same length.
6. Codons may or may not overlap, i.e. their length may be greater than, equal to, or less than the reading restart interval.

Your tasks are:

- A. Using the data and assumptions summarized in (1)-(6) above, explain why
 - coding with doublets is impossible, regardless of reading restart interval.
 - the only possible reading restart interval less than 9 is 3.
- B. From the reasoning in (A), Occam's razor strongly suggests the genetic code is likely to have a reading restart interval of 3, and with (non-overlapping) triplet nucleotide codons. With these assumptions and also assuming that some codons are "stop" codons, construct as much of the genetic code table (which associates triplets of nucleotides to amino-acids or "stop" codons) as the experimental evidence in the *Cracking_the_Code.pdf* document allows you to justify.