

---

Strings X and Y are the same length. What is the relationship between edit distance between X and Y and Hamming distance between X and Y?

- ☒ Edit distance is less than or equal to Hamming distance
- ☐ Edit distance equals Hamming distance
- ☐ Edit distance is greater or equal than Hamming distance
- ☐ Edit distance equals the Hamming distance minus 2

If we're searching for approximate matches of  $P$  within  $T$  allowing up to 4 mismatches, then we should first divide  $P$  up into \_\_\_ partitions:

☐ 8

☐ 2

☐ 4

☒ 5

When we're searching for exact matches for the partitions, which exact matching algorithm do we use?

- ☒ Doesn't matter; any would work
- ☐ Index-assisted exact matching
- ☐ Boyer-Moore
- ☐ Naive exact matching

About \_\_\_\_ of the human genome is covered by Alu repeats

☐ 1%

☒ 10%

☐ 5%

☐ 30%

We can't rely only on dynamic programming alone for the read alignment problem because:

- ☒ The matrix would have too many elements
- ☐ In read alignment, we don't know what labels to give the columns of the matrix
- ☐ Dynamic programming doesn't allow flexible scoring of substitutions and insertions and deletions
- ☐ It can't tell us exactly where an approximate match occurs

The primary use of the index in read alignment is

- ☐ To allow use of a flexible scheme of penalties for substitutions, insertions and deletions
- ☒ To quickly narrow focus to just a few relevant parts of the genome
- ☐ To determine how many mismatches and gaps are in an approximate match

If we have 100 reads, each of length 100, and the genome has length 1000, then the average coverage of the genome is:

☐ 1000

☐ 100

☒ 10

☐ 10000

For  $X, Y$  where  $|X| = |Y|$ , *hamming distance* = minimum # substitutions needed to turn one into the other

For  $X, Y$ , *edit distance* = minimum # edits (substitutions, insertions, deletions) needed to turn one into the other



Which is a true statement: Unlike naive exact matching, Boyer-Moore

- ☐ Is an online algorithm
- ☒ Skips alignments that won't be fruitful
- ☐ Builds a tree representation of the text
- ☐ Fails to find all the matches

A hash collision is an instance here:

- ☒ The key being added falls in the same bucket as a different key added previously
- ☐ The hash function fails to assign a key to a bucket
- ☐ The key being added is equal to a key added previously
- ☐ The key is too large to fit into a single index entry

What is the edit distance between the empty string and a string of length 3?

☐ 0

☒ 3

☐ 1

☐ 4

The purpose of the subsequence-based index is to improve specificity, meaning:

- ☒ A given index hit is more likely to lead to a full match
- ☐ The index will have fewer entries
- ☐ Issuing the same query twice will give the same result
- ☐ More patterns will match the text