1. Algorithms

1.1 Calculating Product's Score

1.1.1 Reduce hardware's score on the scale from 1 to 100

- Define:
  - x : The highest hardware's score in the database.
  - y : The hardware's score in the database.
  - z : The hardware's scor on the scale from 1 to 100 after reduced.
- We have a reduce formular:

$$z = \frac{100}{x} \times y$$

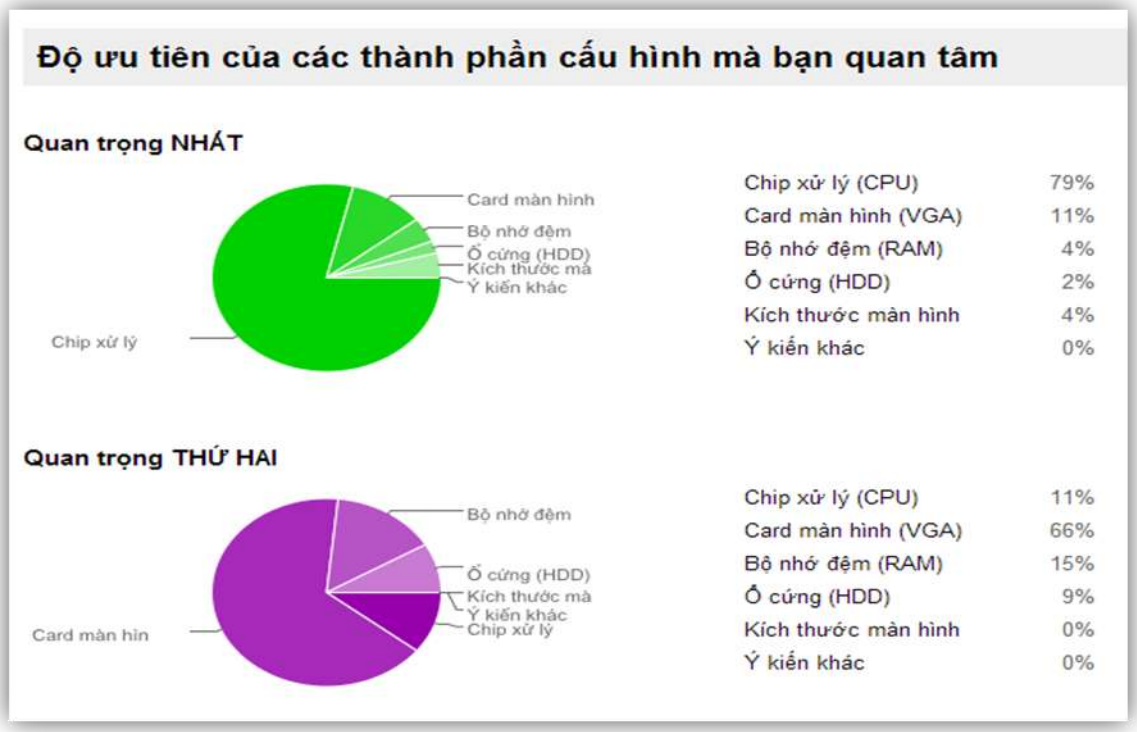
- Example: Reduce the CPU's score on the scale from 1 to 100.
  - Given the highest CPU's score in the database: ***x*** = 13200.
  - Given the CPU's score which we will reduce: ***y*** = 8000.
  - And ***z*** is the CPU's score on the scale form 1 to 100.
  - Apply the reduce formular:

$$z = \frac{100}{13200} \times 8000$$

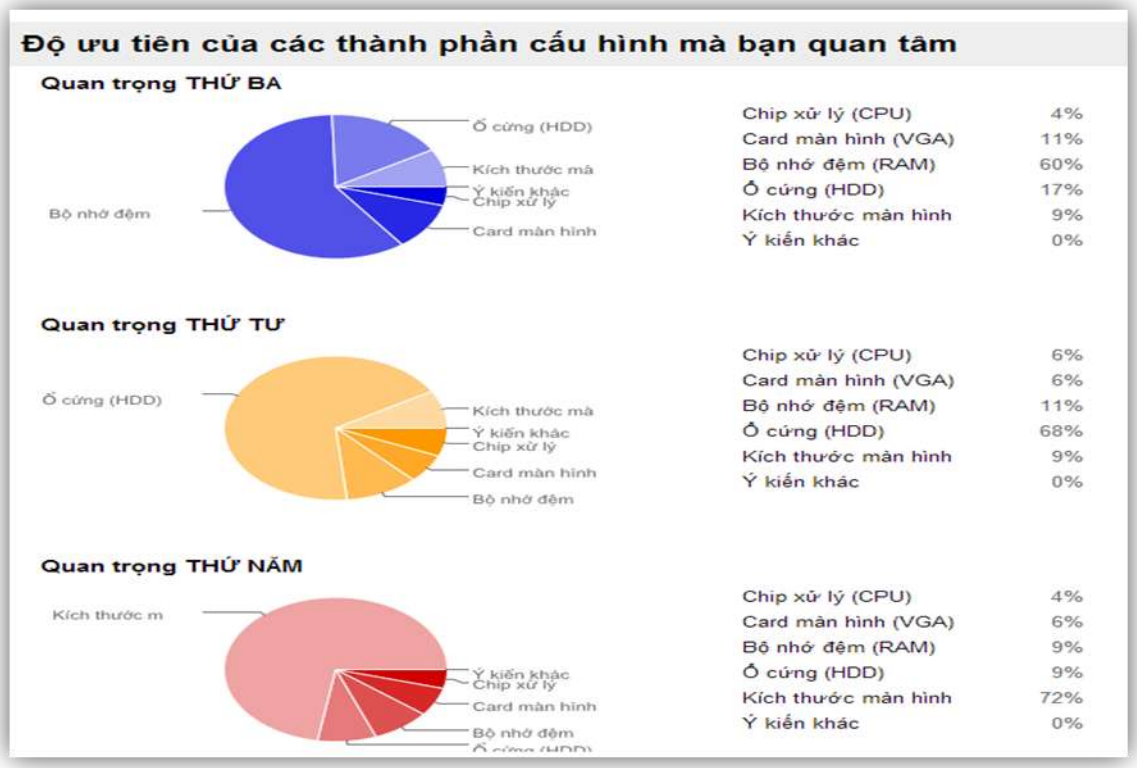
$$z = 60.61$$ (on the scale from 1 to 100).

1.1.2 Calculate product's score and reduce on the scale from 1 to 100

- After calculated all the score of CPU, VGA, RAM, HDD and Display on the scale from 1 to 100, we can calculate the product score.
- Given:
  - c: CPU's score.
  - v: VGA's score.
  - r : RAM's score.
  - h : HDD's score.
  - d : Display's score.
  - P : Product's score.
  - H : The highest product's score in the database.
  - P' : Product's score on the scale from 1 to 100.
- Define priority coefficient:
  - Priority hardwares: 60%



- Normal hardwares: 40%



- Define the formular to calculate product's score:

$$P = \frac{(c + v) \times 60\% + (r + h + d) \times 40\%}{5}$$

- Reduce P on the scale from 1 to 100 by the formular:

$$P' = \frac{100}{H} \times P$$

- Example: Calculate the score of laptop HP Palivion 14 – R006TU with hardware configuration below
    - Given ***c*** : 14.33
    - Given ***v*** : 1.65
    - Given ***r*** : 12.5
    - Given ***h*** : 12.5
    - Given ***d*** : 80.92
    - Given ***H*** : 33.20
- So we have product's score:

$$P = \frac{(14.33 + 1.65) \times 60\% + (12.5 + 12.5 + 80.92) \times 40\%}{5}$$

$$P = 10.39$$

- And reduce product's score on the scale from 1 to 100:

$$P' = \frac{100}{33.20} \times 10.39$$

$$P' = 31.30$$

1.2 String Comparison

1.2.1 Define Problem

Given two strings. Calculate their matching percent.

1.2.2 Requirement

- Robustness to changes of word order: two strings which contain the same words, but in a different order, should be recognised as being similar.
- Language independence: the algorithm should work not only in English, but in many different languages.

1.2.3 Solution

- If a string contains many words, break it into a list of words.
- For each word, we find out how many adjacent character pairs are contained in it.
- Create a function *pairs(s)* which returns a list of adjacent character pairs of string *s*.
- Then, we use below formula to calculate matching percent.

$$similarity(s1, s2) = \frac{2 \times |pairs(s1) \cap pairs(s2)|}{|pairs(s1)| + |pairs(s2)|}$$

1.2.4 Example

Calculate the matching percent of 2 strings: France and French.

- Upper case 2 strings:
    + France → FRANCE.
    + French → FRENCH.
- Break string into list of adjacent character pairs:
    + FRANCE → $\{FR, RA, AN, NC, CE\}$

+ FRENCH → $\{FR, RE, EN, NC, CH\}$

- Calculate its matching percent.

$$similarity(FRANCE, FRENCH) = \frac{2 \times |\{FR, NC\}|}{|\{FR, RA, AN, NC, CE\}| + |\{FR, RE, EN, NC, CH\}|}$$

$$= \frac{2 \times 2}{5 + 5} = \frac{4}{10} = 0.4$$