

1 必修課題レポートの再提出

下記のいずれかが記述されていないレポートは再提出して下さい。

- 1) プログラムの設計方針
- 2) プログラムの大まかな処理の流れ
- 3) 実験で得られたデータの説明
- 4) 考察

注意 考察のないレポートは、調理されていない素材の集まりです。採点しようにも食することはできません。

2 昨年度の必修課題 N-Gram を求めるのに STL を使って書くと

```
#include <iostream>
#include <string>
#include <locale>
#include <utility>
#include <map>
#include <vector>
#include <stdexcept>
#include <algorithm>
using namespace std;

bool compare(const pair<string,int>& x, const pair<string,int>& y)
{
    return x.second > y.second;
}

void output(const pair<string,int>& x)
{
    cout << x.first << "\t: " << x.second << endl;
}

int main (int argc, char* argv[])
{
    try
    {
        cout.exceptions(std::ios::failbit);
        locale::global(locale("jpn"));

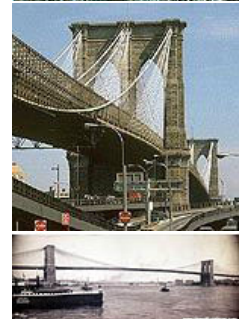
        int n = (argc >=2) ? atoi(argv[1]) : 0 ;
        if (n <= 0) n = 3;

        map<string,int> n_gram;
        string s;
        while(cin >> s)

        for(size_t i = 0; i + n - 1 < s.length(); i++)
            n_gram[s.substr(i,n)]++;

        vector< pair<string,int> > n_gram_sorted(n_gram.begin(), n_gram.end());
        sort(n_gram_sorted.begin(), n_gram_sorted.end(), output);
    }

    catch(exception& e)
    {
        cerr << e.what() << endl;
    }
    return 0;
}
```



3 Little's Law

The average number of things in the system is the product of the average rate at which things leave the system and the average time each one spends in the system. (And if there is a gross 'flow balance' of things entering and leaving, the exit rate is also the entry rate.)

例 60 席あるカフェで、一人 3 時間過ごすとする、あなたの前に並んで待っている人が 30 人いれば、待ち時間は？

4 Some rules of thumb

- 1) π seconds is a nano-century.
- 2) Safety factor is *six times*.

E.g., Suspension bridges: John Roebling designed the Brooklyn Bridge which was built from 1869 to 1883. '*Galloping Gertie*' of the Tacoma Narrows Bridge which tore itself apart in a windstorm in 1940, due to the nonlinearities in aerodynamic lift on suspension bridges modeled by the eddy spectrum. He designed the stiffness of the truss on the Brooklyn Bridge roadway to be *six times* what a normal calculation based on known static and dynamic load would have called for.

5 宿題 — 提出不要

1 空のハッシュ表 (サイズ 19) に次の順でキーが入力される時の linear probing での最終的な格納状況を示せ。

キー: A S E A R C H I N G E X A M P L E
ハッシュ値: 1 0 5 1 18 3 8 9 14 7 5 5 1 13 16 12 5

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18

2 上記と同じハッシュ表で、double hashing での最終的な格納状況を示せ。

キー: A S E A R C H I N G E X A M P L E
ハッシュ値 1: 1 0 5 1 18 3 8 9 14 7 5 5 1 13 16 12 5
ハッシュ値 2: 16 15 12 16 16 14 9 8 3 10 12 10 16 4 1 5 12

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18

3 開番地法 (内部ハッシュ法) で、ハッシュ表の大きさを B 、格納されるデータの個数 M (ただし、 $B \geq M$) とすると、データの挿入の計算手間を求めよ。

6 ACM ICPC に出場した方は、氏名・学籍番号および成績を奥乃まで連絡して下さい。

感想・意見・要望 があれば奥乃まで。(諸君は学ぶ権利があり、教員は分かりやすく教える義務があります。)