

fp_out2ListC. cpp

```

//fp_out2ListC
/* .NET用のinclude//
#include <iostream>
#include <iomanip>
#include <fstream>
//プログラム記述の簡略化//
using namespace std;
*/

//Visual C++6.0用のinclude
#include <iostream.h>
#include <iomanip.h>
#include <fstream.h>
//

//プロトタイプ宣言//
int ceil(int a, int b);
int floor(int a, int b);

// 考えられる列または行を超える数字を入れること //
/*
const int CTTL = 1090000;
const int RTTL = 5000;
*/
const int CTTL = 1100000;
const int RTTL = 5000;

char pb_name[65], unicost, csttype[17];

int rttl, ctll, ww;
int i, j, k;
unsigned int G[CTTL][(RTTL / 32) + 1];
/*
データサイズの計算
CTTL * ((RTTL / 32) + 1) * 4
RTTL = 5000, CTTL = 1090000の時、
1.09M * (5000 / 32 + 1) * 4 = 1.09M * 160 * 4 = 700MB
*/

int cst[CTTL + 10];
double rdensity;
int h[CTTL], rname[200];
/*
rname[]の大きさは、
densityの計算で予測する。
*/

//配列はすべて0スタートで計算//
int main() {
    /*ファイル読み込み. NET用//
    ifstream fin;
    fin.open("fp_out.txt");
    if(fin.fail()) {
        cerr << "Cannot open the ListC file.";
        return 1;
    }
    }

```

fp_out2ListC. cpp

```

ofstream fout;
fout.open("ListC.txt");
if(fout.fail()){
    cerr << "Cannot open the fp_out file.";
    return 2;
}
*/

//ファイル読み込みVisual C++6.0用
ifstream fin("fp_out.txt");
if(!fin){
    cerr << "Cannot open the ListC file.";
    return 1;
}
ofstream fout("ListC.txt");
if(!fout){
    cerr << "Cannot open the fp_out file.";
    return 2;
}

//データ読み込み//
fin >> pb_name >> rttl >> cttl >> csttype >> rdensity;
cout <<"rttl = " << rttl <<" cttl = " << cttl << endl;

//cost[]の読み込み//
for( i = 0 ; i < ceil(cttl , 10) ; i++){
    fin >> cst[(10 * i) + 0] >> cst[(10 * i) + 1]
        >> cst[(10 * i) + 2] >> cst[(10 * i) + 3]
        >> cst[(10 * i) + 4] >> cst[(10 * i) + 5]
        >> cst[(10 * i) + 6] >> cst[(10 * i) + 7]
        >> cst[(10 * i) + 8] >> cst[(10 * i) + 9];
}

//G[]の読み込み//
for( i = 0 ; i < cttl ; i++)
    for( j = 0 ; j < ceil(rttl , 32) ; j++)
        fin >> G[i][j];

cout <<"An output working start to ListC." << endl;

//fp_outからListCへのデータ変換作業//
fout <<" " << rttl <<" " << cttl << endl;
for( i = 0; i < cttl; i++ ){
    h[i] = 0;
    for( j = 0; j < rttl; j++ ){
        ww = G[i][floor(j, 32)];
        ww = (ww << (j % 32));
        if(ww < 0){
            //ListCは1スタートで番号を保存しているためj+1を
            rname[h[i]] = j + 1;
            h[i]++;
        }
    }
    fout <<" " << cst[i] <<" " << h[i];
    for( k = 0 ; k < h[i] ; k++) {

```

代入//

```

        fp_out2ListC.cpp
    fout << " " << rname[k];
    }
    fout << endl;
}

cout << "An output working end to ListC." << endl;

//ファイルクローズ//
fin. close();
fout. close();
}
//プロトタイプ宣言の計算式//
int ceil(int a, int b){
    if(a % b)
        return (a / b) + 1;

    return a / b;
}
int floor(int a, int b){
    return a / b;
}

```