

Δομές Δεδομένων και Αλγόριθμοι

Αλγόριθμοι ωμής βίας (V1.0)

Χρήστος Γκόγκος

Πανεπιστήμιο Ιωαννίνων, Τμήμα Πληροφορικής και Τηλεπικοινωνιών (2019-2020)

- Η ωμή βία ως στρατηγική αντιμετώπισης αλγοριθμικών προβλημάτων είναι μια απευθείας προσέγγιση επίλυσης η οποία στηρίζεται άμεσα στη διατύπωση του προβλήματος και τη διάθεση υπολογιστικής ισχύος.
- Συνήθως αποτελεί τον “προφανή” τρόπο για την επίλυση ενός προβλήματος αλλά σπάνια ο τρόπος αυτός είναι αποδοτικός.
- Μπορεί να χρησιμοποιηθεί για μικρά μεγέθη προβλημάτων και η επίλυση με ωμή βία να αποτελέσει μέτρο σύγκρισης για άλλες, εναλλακτικές μεθόδους επίλυσης.
- Μια πρώτη εφαρμογή της προσέγγισης ωμής βίας μπορεί να έχει ως αποτέλεσμα έναν αλγόριθμο ο οποίος μπορεί να βελτιωθεί απαιτώντας σχετικά μικρή επιπλέον προσπάθεια.

Ταξινόμηση με επιλογή (selection sort)

Στην ταξινόμηση με επιλογή εντοπίζουμε στην αρχική λίστα την μικρότερη τιμή και την εναλλάσσουμε με το πρώτο στοιχείο της λίστας. Στη συνέχεια εξετάζουμε τη λίστα ξεκινώντας από το δεύτερο στοιχείο της ξανά για να εντοπίσουμε το μικρότερο στοιχείο και το εναλλάσσουμε με το δεύτερο στοιχείο της λίστας. Μετά από τόσα περάσματα, όσα το μέγεθος της λίστας μείον ένα, η λίστα θα είναι πλέον ταξινομημένη.

Ταξινόμηση με επιλογή (κώδικας)

```
#include <iostream>
#include <algorithm> //swap
using namespace std;

void selection_sort(int a[], int n) {
    for (int i = 0; i < n - 2; ++i) {
        int pmin = i;
        for (int j = i + 1; j < n - 1; ++j)
            if (a[j] < a[pmin])
                pmin = j;
        swap(a[i], a[pmin]);
    }
}

int main() {
    int a[] = {3, 5, 1, 7, 8, 9};
    int n = sizeof(a) / sizeof(int);
    selection_sort(a, n);
    for (int i = 0; i < n; i++)
        cout << a[i] << " ";
    cout << endl;
}
```

Ζητείται η εύρεση των δύο πλησιέστερων σημείων για ένα σύνολο σημείων για τα οποία γνωρίζουμε τις συντεταγμένες τους στο καρτεσιανό επίπεδο. Η απόσταση δύο σημείων $P_i = (x_i, y_i)$ και $P_j = (x_j, y_j)$ είναι η Ευκλείδεια απόσταση $d(P_i, P_j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$

Η προσέγγιση ωμής βίας οδηγεί στον αλγόριθμο υπολογισμού της απόστασης για κάθε πιθανό ζεύγος σημείων και στην εύρεση του ζεύγους με την μικρότερη απόσταση. Για να αποφευχθεί η εξέταση του ίδιου ζεύγους δύο φορές εξετάζονται μόνο ζεύγη σημείων (P_i, P_j) για τα οποία ισχύει $i < j$

```
#include <iostream>
#include <ctime> // time
#include <limits> // numeric_limits<double>::max()
#include <cmath> //sqrt, pow
#define N 10000
using namespace std;
int main() {
    double x[N], y[N]; srand(time(NULL));
    for (int i = 0; i < N; ++i) {
        x[i] = (double)rand() / (double)RAND_MAX;
        y[i] = (double)rand() / (double)RAND_MAX;
    }
    int index1, index2; double dmin = numeric_limits<double>::max();
    for (int i = 0; i < N - 1; ++i)
        for (int j = i + 1; j < N; ++j) {
            double d = sqrt(pow((x[i] - x[j]), 2)
                + pow((y[i] - y[j]), 2));
            if (d < dmin) { dmin = d; index1 = i; index2 = j; }
        }
    cout << "Point1=(" << x[index1] << "," << y[index1] << ")" << endl;
    cout << "Point2=(" << x[index2] << "," << y[index2] << ")" << endl;
    cout << "Distance=" << dmin << endl;
}
```