

Άσκηση 8.4



Δεδομένα που αφορούν εργαζόμενους αποθηκεύονται σε πίνακα με όνομα `factory`, που ορίζεται σύμφωνα με τα παρακάτω :

```
struct address {
    int city_code;
    int street_code
    int number;
};
struct person {
    int code;
    struct address addr;
    float salary;
};
struct person factory[10];
```

Άσκηση 8.4



Να γραφεί πρόγραμμα σε γλώσσα C που :

1. θα καταχωρεί τα στοιχεία των εργαζομένων στον πίνακα `factory`.
2. θα βρίσκει και θα εκτυπώνει τα στοιχεία του εργαζόμενου με τον μεγαλύτερο μισθό με χρήση απλών μεταβλητών.
3. θα βρίσκει και θα εκτυπώνει τα στοιχεία του εργαζόμενου με τον μεγαλύτερο μισθό χρησιμοποιώντας τις παρακάτω δηλώσεις :

```
float most=factory[0].salary;  
struct person *aux, *who;
```

Άσκηση 8.4 - Δομές



```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#define N 10

struct address {
    int city_code;
    int street_code;
    int number;
};
struct person {
    int code;
    struct address addr;
    float salary;
};
struct person factory[N];
```

Άσκηση 8.4 - data_entry



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
void data_entry() {  
  
    for (int i = 0; i < N; i++) {  
        factory[i].code = 1000 + i;  
        factory[i].addr.city_code = rand() % 100;  
        factory[i].addr.street_code = rand() % 1000;  
        factory[i].addr.number = rand() % 200;  
        factory[i].salary = 1000.0f + (rand() % 5000); // 1000-5999  
    }  
}
```

Άσκηση 8.4 - print_person



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
void print_person(struct person p) {  
  
    printf("Κωδικός: %d\n", p.code);  
    printf("Διεύθυνση: Πόλη %d, Οδός %d, Αριθμός %d\n",  
           p.addr.city_code, p.addr.street_code, p.addr.number);  
    printf("Μισθός: %.2f\n", p.salary);  
}
```

Άσκηση 8.4 - main (1/3)



```
int main() {
    srand(time(0));

    data_entry();

    printf("ΣΤΟΙΧΕΙΑ ΟΛΩΝ ΤΩΝ ΕΡΓΑΖΟΜΕΝΩΝ\n");
    for (int i = 0; i < N; i++) {
        printf("Εργαζόμενος %d:\n", i + 1);
        print_person(factory[i]);
    }
}
```

Άσκηση 8.4 - main (2/3)



```
// --- Μέρος 1: Με απλές μεταβλητές ---  
float max_salary = factory[0].salary;  
int max_index = 0;  
for (int i = 1; i < N; i++) {  
    if (factory[i].salary > max_salary) {  
        max_salary = factory[i].salary;  
        max_index = i;  
    }  
}  
  
printf("\nΜεγαλύτερος μισθός (με απλές μεταβλητές)\n");  
print_person(factory[max_index]);
```

Άσκηση 8.4 - main (3/3)



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
// --- Μέρος 2: Με χρήση δεικτών ---  
float most = factory[0].salary;  
struct person *aux;  
struct person *who = &factory[0];  
  
for (aux = factory + 1; aux < factory + N; aux++) {  
    if (aux->salary > most) {  
        most = aux->salary;  
        who = aux;  
    }  
}  
printf("\nΜεγαλύτερος μισθός (με δείκτες)\n");  
print_person(*who);  
return 0;  
}
```