

## Oefeningen

### Oefeningen sectie 1.12 - Records in DRAMA

1. Vertaal het volgende C programma.

```
struct rationaalgetal
{
    int teller;
    int noemer;
    int gehdeel;
};

/* =====
 * Eerder vertaald:
 * ===== */
struct rationaalgetal maakrationaal(int gehdeel, register int teller,
                                     int noemer) { ... } // zie huistaak
void drukrationaal (struct rationaalgetal g) { ... } // zie huistaak van vorige week
int kgv(int a, register int b) // kleinst gemeen veelvoud ==> Zie oefenzitting 3
int ggd(register int a, int b) // grootst gemene deler ==> zie oefenzitting 3
// =====

struct rationaalgetal verschilrationaal (register struct rationaalgetal * a,
                                         struct rationaalgetal b)
{
    struct rationaalgetal verschil;
    register int kv, fac;
    // trek gehele delen van elkaar af
    verschil.gehdeel = a->gehdeel - b.gehdeel;

    // noemer is kleinste gemeen veelvoud van de 2 noemers
    kv = kgv(a->noemer, b.noemer);
    verschil.noemer = kv;

    // teller is het verschil van de tellers (vermenigvuldigd met een factor)
    fac = kv / a->noemer;
    verschil.teller = fac * a->teller;
    fac = kv / b.noemer;
    verschil.teller -= fac * b.teller;

    // vereenvoudig teller / noemer
    fac = ggd (verschil.teller, verschil.noemer);
    if (fac != 1) {
        verschil.teller /= fac;
        verschil.noemer /= fac;
    }
}
```

```

    return (verschil);
}

struct rationaalgetal x; // globale variabele

main ()
{
    struct rationaalgetal y, vs;

    x = maakrationaal(10, 7, 8);
    y = maakrationaal(5, 1, 3);

    vs = verschilrationaal (&x, y);

    drukrationaal(vs); // 10 7/8 - 5 1/3 = 5 7/12
}

```

## 2. Vertaal het volgende C-programma

```

struct rec {
    int len;
    int rij[5];
};

struct rec mkRec (int * r, int n)
{
    struct rec tmp;
    register int i;

    tmp.len = n;
    for (i = 0; i < n; i++)
        tmp.rij[i] = r[i];

    return (tmp);
}

void rec2Rij (struct rec s, register int r[])
{
    register int i, * ptr;

    ptr = s.rij;
    for (i = 0; i < s.len; i++)
        r[i] = *ptr++;
}

int grij[] = {0, 0, 0, 0, 0};

```

```

main ()
{
    int rij[] = {1, 2, 3, 4, 5};
    struct rec REC;
    register int i;

    REC = mkRec(rij, 3);
    rec2Rij (REC, grij);

    for (i = 0; i < 5; i++)
        printint(grij[i]);
}

```

3. Bestudeer het volgende C-programma. Waarom loopt het fout?

```

struct loc {
    int x, y;
}

struct loc * mkLoc (int a, int b)
{
    struct loc l;

    l.x = a;
    l.y = b;

    return (&l);
}

void druk (struct loc l)
{
    printint (l.x, l.y);
}

main ()
{
    struct loc mijnLoc, * wzLoc;

    mijnLoc.x = 10;
    mijnLoc.y = 20;
    wzLoc = mkLoc (2, 3);

    druk (mijnLoc);
    ...
    mijnLoc = *wzLoc;
    druk (mijnLoc); // Zou 2 3 moeten afdrukken!
}

```