

Oefeningen sessie 5 - Records - versie 2

Oefeningen

Oefeningen sectie 1.12 - Records in DRAMA

1. Gegeven:

```
struct boek {
    int isbn;           // het isbn nummer van het boek
    int uitgeleend;     // uitgeleend (true) of in de bib (false)
    int numbers[20];    // numbers van de huidige en vorige leners
    int uitleningen;    // het aantal uitleningen
    int staat;         // de staat van het boek
};

struct lener {
    int nummer;
    int uitgeleend[5]; // isbn numbers van de uitgeleende boeken
    int aantal;        // het aantal uitgeleende boeken
};

struct boek bib[1000];
struct lener lezers[500];
struct boek bib[1000];
struct lener lezers[500];
// Declaraties van globale variabelen: wzBoek, ptrLezer, idxBoek, idxLezer

main ()
{
    wzBoek = &boek[idxBoek];

    wzBoek->numbers[wzBoek->uitleningen] = lezers[idxLezer].nummer;
    wzBoek->uitleningen += 1;

    ptrLezer = &lezers[idxLezer];
    ptrLezer->uitgeleend[ptrLezer->aantal] = wzBoek->isbn;
    ptrLezer->aantal++;
}
```

- o Vertaal het hoofdprogramma als de variabelen als volgt gedeclareerd zijn:
(variabelen die niet in registers zitten, zitten in het geheugen met als symbolisch adres hun naam).

// Declaratie A	// Declaratie B
int idxBoek;	int idxBoek;
register int idxLener;	register int idxLener;
struct boek *wzBoek;	register struct boek * wzBoek;
register struct lezer *ptrLezer	struct lezer *ptrLezer;

2. Vertaal het volgende C programma.

```
struct student {
    int studNr;
    int faculteit;
    int richting;
    int faciliteit;
};

struct examen {
    int vakNr;
    int datum;
    int duur;
    struct student *student;
};

struct examen examens[1000];
struct student studenten[200];

main ()
{
    struct student stud;
    register struct student* wzStud;
    register int stdIdx, exIdx;

    stud.faculteit = 1;
    stud.richting = 2;
    stud.optie = 3;

    ... // stdIdx en exIdx krijgen een waarde
}
```

```

    studenten[stdIdx] = stud;
    examens[exIdx].student = &studenten[stdIdx];

    ...

    examens[exIdx].duur = 3;
    if (examens[exIdx].student->faciliteiten)
        examens[exIdx].duur += 1;    // 1 uur extra tijd

    ...

    wzStudent = examens[exIdx - 5].student;
    wzStudent->richting = 3;
}

```

3. Vertaal het volgende C-programma. Teken de stapel zoals ze er uit ziet net vóór de terugkeer uit de functie vereenvoudig.

```

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

int gcd (register int a, int b)
{
    if (b == 0)
        return a;
    return gcd (b, a % b);
}

void vereenvoudig (struct rationaalgetal in, register struct rationaalgetal * uit)
{
    register int deler;

    deler = gcd (in.teller, in.noemer);
    in.teller /= deler;
    in.noemer /= deler;
    uit->gehdeel = in.gehdeel + (in.teller / in.noemer);
    uit->teller = in.teller % in.noemer;
    uit->noemer = in.noemer;
}

void leesrationaal (struct rationaalgetal * g) { ... } // Zie huistaak
void drukrationaal (struct rationaalgetal g) { ... } // Zie huistaak

main ()
{
    struct rationaalgetal q, r;

    leesrationaal (&q);
    while (q.noemer != 0)
    {
        vereenvoudig (q, &r);
        drukrationaal (r);
        leesrationaal (&q);
    }
}

```

4. Vertaal het volgende C-programma.

Teken de stapel zoals ze eruit ziet net vóór de terugkeer uit de functie cogr.

```

struct Tco
{
    int x, y;
};

struct Tco corij[50];

int cogr (struct Tco co1, register struct Tco * co2wz)
{
    if (co1.x * co1.x + co1.y * co1.y >
        co2wz->x * co2wz->x + co2wz->y * co2wz->y)
        return (1);
    else
        return (0);
}

void leesco (struct Tco * co)
{
    co->x = getint();
    co->y = getint();
}

void drukco (struct Tco co)
{

```

```

    printint (co.x, co.y);
}

struct Tco * vindMaxCo (struct Tco cr[], int n)
{
    register int i, max; ik zit boven btw

    max = 0;
    for (i = 0; i < n; i++)
    {
        if ((cogr(cr[i], &cr[max]) == 1)
            max = i;
    }
    return &cr[max]; // teruggeefwaarde (= adres) via R0
}

main()
{
    struct Tco * wz;
    register int i;
    int aantal;

    aantal = getint();

    for (i=0; i<aantal; i++)
        leesco (&corij[i]);

    wz = vindMaxCo(corij, aantal);

    drukco (*wz); // *wz !!!
}

```

5. Vergelijk de volgende 2 C-programma's. Welke waarde zal er afgedrukt worden in beide gevallen?

- Verklaar het verschil!
- Teken de volledige stapel zoals ze er uit ziet net na de toekenning ' $s^2 = 10$ ' in de procedure 'wijzig'.

```

int r[] = {1, 2, 3, 4, 5};

void wijzig (int s[])
{
    s[2] = 10;
}

main ()
{
    wijzig(r);
    printint(r[2]);
}

```

```

struct rec {
    int r[5];
};

int r[] = {1, 2, 3, 4, 5};
struct rec rr;

void wijzig (struct rec s)
{
    s.r[2] = 10;
}

main ()
{
    register int i, * wz;

    // kopieer de rij r naar de rij rr.r

    for (i = 0, wz = r; wz < &r[5]; )
        rr.r[i++] = *wz++;

    wijzig(rr);

    printint(rr.r[2]);
}

```