

Oefenzitting 8 - Lijsten

Oefening 1

Vertaal het volgende C programma.

```
struct elem {
    int info;
    struct elem * volgend;
    struct elem * vorig;
};
// Opbouwen van een dubbelgelinkte circulaire lijst met kopelement
main() {
    struct elem * lst;
    register int getal;
    register struct elem * ptr, * n;

    lst = (struct elem *) alloc (sizeof(struct elem)); // kopelement
    lst->info = 0;
    lst->volgend = lst->vorig = lst;

    getal = getint();
    while (getal > 0) {
        // nieuw toe te voegen element
        n = (struct elem *) alloc (sizeof(struct elem));
        n->info = getal;
        // achteraan toevoegen
        lst->vorig->volgend = n;
        n->vorig = lst->vorig;
        lst->vorig = n;
        n->volgend = lst;
        getal = getint();
    }
    // getallen zoeken in lijst
    getal = getint();
    while (getal > 0) {
        lst->info = getal; // kopelement = schildwacht
        // leeg lichaam for-lus!
        for (ptr = lst->volgend; getal != ptr->info; ptr = ptr->volgend);

        if (ptr == lst) {
            printint(getal, 0); // niet aanwezig in lijst
        } else {
            printint(getal, 1); // aanwezig in lijst
        }
    }
}
```

```

    }
    getal = getint();
}
}

```

Oefening 2

Vertaal de functie druklst die een dubbelgelinkte circulaire lijst afdrukt (*voorwaarts* indien voorw = 1, of *van achter naar voor* indien voorw = 0).

```

void druklst (struct elem * lst, int voorw) {
    register struct elem * ptr;

    if (voorw == 1)
        for (ptr = lst->volgend; ptr != lst; ptr = ptr->volgend)
            printint(ptr->info);
    else
        for (ptr = lst->vorig; ptr != lst; ptr = ptr->vorig)
            printint(ptr->info);
}

```

Roep deze functie op in het hoofdprogramma van de vorige oefening:

```

...
main() {
    ...
    druklst(lst, 1);
    druklst(lst, 0);
    // Zoeken in de lijst
    ...
}

```

Oefening 3

Wat is de vertaling van het volgende C-programma? Het is de iteratieve versie van de torens van Hanoi. De lijst houdt de verplaatsingen bij die nog moeten uitgevoerd worden; in elke iteratie wordt het element dat vooraan in de lijst staat uit de lijst gehaald en behandeld:

- ofwel betreft het de verplaatsing van 1 ring, en dan wordt dit afgedrukt
- ofwel betreft het de verplaatsing van meerdere ringen (vb. A ringen van VAN naar NAAR): in dit geval worden er 3 elementen *VOORAAN* aan de lijst toegevoegd: (A-1 ringen van VAN naar VIA), (1 ring van VAN naar NAAR), (A-1 ringen van VIA naar NAAR).

Er is een functie voorzien (`nieuw`) die een nieuw element alloceert en initialiseert; m.o. dat deze functie ook het element vooraan aan een lijst toevoegt en de nieuwe lijst als resultaat teruggeeft.

```
struct verpl {
    int van;
    int naar;
    int aantal;
    struct verpl * volg;
};
int aNieuw = 0; // aantal gecreeerde elementen op de heap

struct verpl * nieuw (int van, int naar, int aantal,
                      register struct verplaatsing * lijst) {
    register struct verpl * v = (struct verpl *) alloc(sizeof(struct verpl));
    aNieuw++;
    v->van = van;
    v->naar = naar;
    v->aantal = aantal;
    v->volg = lijst;    // plaats het element vooraan in de lijst
    return (v);
}
main () {
    int aantal;
    register struct verpl * lijst;
    register struct verpl * huidig;
    int via;
    int aVerpl; // aantal verplaatsingen
    aantal = getint();
    // verplaats AANTAL ringen van 1 naar 2
    lijst = nieuw(1, 2, aantal, NULL);
    aVerpl = 0;
    aNieuw = 1;
    while (lijst != NULL) {
        huidig = lijst;
        lijst = lijst->volg; // haal element uit de lijst
        if (huidig->aantal == 1) {
            printint(huidig->van, huidig->naar);
            aVerpl++;
        } else {
            via = 6 - huidig->van - huidig->naar;
            /* genereer in omgekeerde volgorde en voeg vooraan toe aan lijst */
            lijst = nieuw (via, huidig->naar, huidig->aantal - 1, lijst);
            lijst = nieuw (huidig->van, huidig->naar, 1, lijst);
        }
    }
}
```

```
        lijst = nieuw (huidig->van, via, huidig->aantal - 1, lijst);
    }
}
printint(aVerpl, aNieuw);
}
```

Kan je het aantal verplaatsingen (`aVerpl`) en het aantal gecreerde records (`aNieuw`) in een formule in functie van het aantal schijven schrijven?