

Oefenzitting 2: Rijen en matrices

Oefeningen sectie 1.7 - Werken met rijen in Drama

1. Gegeven het volgende C programma:

```
int a[] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
int i, j, som, idx;

main()
{
    for (i=0; i<5; i++) {
        idx = getint();
        som = 0;
        for (j=0; j<tot; j++)
            som += a[j];
        printint(som);
    }
}
```

- Vertel in woorden wat dit programma doet.
- Vertaal het volgende C programma naar DRAMA door de adreswijziging via het rekenen met bevelen te laten gebeuren. (Dus zonder gebruik te maken van indexregisters).
- Doe hetzelfde maar nu met index-registers.

2. Variant van oef 2 uit de cursus. Bereken de eerste n Fibonacci-getallen en bewaar ze in een rij.

De getallen van Fibonacci worden gedefinieerd als: $F[i] = F[i-1] + F[i-2]$ waarbij $F_0 = F_1 = 1$. Druk deze n Fibonacci getallen ook af. Veronderstel dan n maximaal 100 is. Vertaal het volgende C-programma:

```
int f[100], i, n;

main()
{
    n = getint();
    f[0] = 1;
    f[1] = 1;
    printint (f[0], f[1]);
    for (i = 2; i < n; )
    {
        f[i] = f[i-1] + f[i-2];
        printint(f[i++]);
    }
}
```

3. Vertaal het volgende C-programma, waarbij de wijzer **ptr** in het geheugen bewaard wordt en **qptr** in **R5**.

```
int * ptr, * qptr; // bewaar ptr in het geheugen en qptr in R5
int a[100];
int n, i; // bewaar i in R1 en n in R2

main()
{
    ... // inlezen van a

    n = ...;
    i = ...;

    ptr = &a[i];
    ...
    *ptr++ = a[2*i-3] + a[4];
    ...
    qptr = &a[50-i] - 2;
    ...
    *qptr++ = a[a[n] - 3] - 4;
    ...
}
```

4. Bestudeer het volgende Drama-programma:

```
          SPR MAIN
X:        RESGR 1
Y:        RESGR 10

MAIN:     LEZ
          BIG    R0,X
          HIA.w  R1,0

WHILE:    VGL     R1,X
          VSP     GRG,STOP
          LEZ
          HIA     R2,R0
          MOD.w  R2,2
          VSP     NUL,EIF
          OPT.w  R0,1
EIF:      DRU
          BIG    R0,Y(R1+)
          SPR    WHILE

STOP:     STP
          EINDPR
```

- Beschrijf wat dit programma doet.
- Hoeveel instructies worden er uitgevoerd als de invoer bestaat uit: 3, 17, 2, 5, 105, 20
- Hoeveel geheugentoeegangen waren er hiervoor nodig? Maak een onderscheid tussen een leestoeegang en een schrijftoeegang.
(Hint: om een instructie te kunnen uitvoeren moet deze ook eerst opgehaald worden)

5. Gegeven het volgende C programma.

```
int a, *ptr, *qptr;
int b[10];

main()
{
    for (ptr = &b[9]; ptr >= b; )
        *ptr-- = getint();

    ptr = &b[4];

    qptr = &b[3] + 2;

    printing(*ptr, *qptr);
}
```

Verbeter alle fouten in het overeenkomstige DRAMA programma. De plaats waar de variabelen bewaard worden mag niet gewijzigd worden.

```
| a    in R1
| ptr  in R2
| qptr in geheugen
| b in geheugen

FOR:   HIA.a R2,B+9
      VGL   R2,B
      LEZ
      BIG.i R0,0(-R2) | *ptr-- = getint()
      SPR   FOR

      | ptr = &b[4]
      HIA   R2,B+4

      | qptr = &b[3] + 2
      HIA.a R0,B+3
      OPT   R0,2
      BIG   R0,QPTR

      | printing(*ptr + *qptr);
      HIA.i R0,0(R2)
      OPT   R0,QPTR
      DRU
      MOD.w R2,2
      VSP   NUL,EIF
      OPT.w R0,1

      STP
QPTR: RESGR 1
B:    RESGR 1

EINDPR
```

Oefeningen sectie 1.8 – Meerdimensionale rijen

1. Oef 8 uit cursus

Zij gegeven een matrix A met n rijen en m kolommen ($n \leq 20$ en $m \leq 20$). Gevraagd wordt deze matrix een kwartslag in tegenwijzerzin te roteren en het resultaat weg te bergen in een $m \times n$ matrix B en deze matrix af te drukken. De invoer bestaat uit n , m , en daarna de $n \times m$ elementen van A (rij per rij ingegeven).

```
int a[20][20], b[20][20];
int i, j, n, m;
main()
{
    n = getint();
    m = getint();
    ... // inlezen van de 2-dim tabel a

    /* kwartslag roteren */
    for (j=0; j<m; j++)
        /* de elementen van kolom j van a worden geplaatst in
           rij (m-1)-j van b */
        for (i=0; i<n; i++)
            b[m-1-j][i] = a[i][j];

    ... // afdrukken van het resultaat
}
```

- Lineariseer bovenstaand programma (kolom linearisatie). Vertaling naar DRAMA is niet nodig.

2. Oefening

```
int matrix[10][20];
main()
{
    ...
    n = getint();

    for (i = 5; i < n; i++)
        for (j = i-1; j >= 4; j--)
            matrix [2*i - j][- 15*i + 20*j + n] = i+j;
    ...
}
```

- Lineariseer bovenstaand programma (rij linearisatie). Vertaling naar DRAMA is niet nodig.
- Voor welke waarden van n wordt het programma correct uitgevoerd?