

Uitwerking proeftoets ME-INPROG1-11

Opdracht 1

```
#include <stdio.h>
#include <ctype.h>

int main(void)
{
    int ch;
    FILE *ofp;

    ofp=fopen("c:/windows/temp/hoofdletter.txt","w");

    do {
        ch=getchar();
        putc(toupper(ch),ofp);
    } while(ch != '\n');

    fclose(ofp);
}
```

Opdracht 2

```
#include <stdio.h>
#include <stdlib.h>

int main(void)
{
    int i, j, getal;
    FILE *ifp, *ofp;

    if ( (ifp=fopen("c:/windows/temp/getalmat.txt","r")) == NULL )
    {
        printf("Kan c:/windows/temp/getalmat.txt niet openen\n");
        exit (-1);
    }
    ofp=fopen("c:/windows/temp/difmat.txt","w");

    for(i=0;i<11;++i)
    {
        for(j=0;j<11;++j)
        {
            fscanf(ifp,"%d",&getal);
            fprintf(ofp,"%5d",10-getal);
        }
        fprintf(ofp,"%c",' \n');
    }
    fclose(ifp);
    fclose(ofp);
}
```

Opdracht 3

```
#include <stdio.h>

// functie prototyping
void druk_af(void);

int main(void)
{
    printf("Voer een regel in: ");
    druk_af();
    printf("\n\n");

    getchar();
}

void druk_af(void)
{
    int c; // de lokale c zorgt voor tijdelijke opslag
           // deze wordt namelijk op de STACK geplaatst

    if((c=getchar()) != '\n') // stop conditie !
        druk_af();           // roept zichzelf aan !

    putchar(c);
}
```

Opdracht 4 (met een while en een for)

```
#include <stdio.h>

int main(void)
{
    int teller, ch[255];

    // maximale regelomvang windows en unix is 255 karakters
    // ch[] is nodig voor tijdelijke opslag van de gelezen karakters
    // ch is van type int omdat getchar EN putchar int verwachten !

    // inlezen regel
    teller=0;
    while((ch[teller++]=getchar()) != '\n');

    // teller wijst nu naar het array element NA de \n !!!! ofwel 2 te ver

    // omgekeerd afdrukken regel
    for(teller-=2; teller>=0;teller--)
        putchar(ch[teller]);
}
```

Opdracht 4

Deze opdracht kan ook met alleen for's worden opgelost !

Let op dat teller nu slechts 1 positie te ver wijst !

```
#include <stdio.h>

int main(void)
{
    int teller, ch[255];

    // maximale regelomvang windows en unix is 255 karakters
    // ch[] is nodig voor tijdelijke opslag van de gelezen karakters
    // ch is van type int omdat getchar EN putchar int verwachten !

    // inlezen regel
    for(teller=0; (ch[teller++]=getchar()) != '\n'; teller++);

    // teller wijst nu naar het array element van de \n !!!! ofwel 1 te ver
    // dit heeft te maken met het moment waarop de teller wordt verhoogd.

    // omgekeerd afdrukken regel
    for(teller-=1; teller>=0;teller--)
        putchar(ch[teller]);
}
```

Opdracht 4 (met tweemaal while)

Dat kan dus ook.

Alle drie de oplossingen zouden bij een juiste werking worden goedgekeurd.

Ook oplossingen met een of twee **do {} while()** kunnen correct werken.