Übung 5 - Lösung

1. Arrays und Schleifen

Erstellen Sie ein Programm das ab der Speicheradresse 'POS_ZAHL' die ersten 10 positiven und ab der Speicheradresse 'NEG_ZAHL' die ersten 10 negativen INTEGER-Zahlen erzeugt. Die 0 gehört zu den positiven Zahlen.

a) Erstellen Sie ein Flussdiagramm für den Algorithmus

b) Erstellen Sie das Assembler Program

Folgendermaßen können Arrays definiert werden:
pos_zahl DD 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
neg_zahl DD 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0

Der Zugriff erfolgt jeweils über die Adresse und den Verschiebeanteil in Byte:
z.B. MOV [neg_zahl+8], EAX

Geben Sie die Zahlen per printf - Funktion aus.

```
%include "io.inc"
section .data
pos_zahl DD 0,0,0,0,0,0,0,0,0,0
neg_zahl DD 0,0,0,0,0,0,0,0,0,0
format DB "%d, %d, %d, %d, %d, %d, %d, %d, %d, %d",0xA,0

section .text
global CMAIN
extern printf
CMAIN:
mov ebp, esp; for correct debugging
;write your code here

mov eax, 0
mov ebx, 0
mov ecx, 10
mov edx, 0

Schleife:
mov [pos_zahl+ebx], edx
mov [neg_zahl+ebx], eax
add ebx, 4
inc edx
dec eax
loop Schleife
```

```
; Ausgabe pos_zahlen
mov ecx, 10
mov ebx, 36
Schleife2:
push dword [pos_zahl+ebx]
sub ebx, 4
loop Schleife2

push format
call printf
mov esp, ebp

; Ausgabe neg_zahlen
mov ecx, 10
mov ebx, 36
Schleife3:
push dword [neg_zahl+ebx]
sub ebx, 4
loop Schleife3

push format
call printf
mov esp, ebp

xor eax, eax
ret
```

2. Makros und Unterprogramme

a) Wenn Sie ein möglichst schnelles Assembler Programm schreiben wollen, sollten Sie dann Makros oder Unterprogramme verwenden? (mit Begründung)

Unterprogramme sollten nicht verwendet werden, da bei einem Aufruf die die Rücksprungadresse auf dem Stack gespeichert werden muss und dann wieder zurück. Dies verbraucht zusätzliche Rechenzeit.

Makros können verwendet werden, da hier der Code während des Kompilieren ersetzt wird. Zur Programmlaufzeit werden damit keine zusätzlichen Befehle ausgeführt.

b) Können einem Unterprogramm Parameter übergeben werden?

Nein, hier muss ein Umweg über die Register oder den Stack verwendet werden.