

Stapel und Unterprogramme:

1. Schreiben Sie ein Unterprogramm „HexEin“, welches ein übergebenes ASCII-Zeichen der Form '0'...'9','A'...'F' in eine Zahl mit $0 \leq \text{Zahl} \leq 15$ umwandelt.

- a) Die Übergabe des Zeichens soll in AH, die Rückgabe in AL erfolgen.
- b) Die Übergabe und die Rückgabe soll über denselben Stapelspeicherplatz erfolgen.
- c) Die Übergabe soll über den Stapel, die Rückgabe über AL erfolgen. Nach Beendigung des Unterprogramms soll der Stapel abgeräumt sein.

2. Für die Ermittlung des größten gemeinsamen Teilers zweier Zahlen A und B haben wir in Klasse 11 den Euklidischen Algorithmus kennengelernt. Eine besonders einfache Version ist durch die folgende Fallunterscheidung gegeben:

$$\text{GGT}(A,B) = \begin{cases} \text{GGT}(A-B,B) & , \text{ falls } A > B \\ \text{GGT}(B,A) & , \text{ falls } A < B \\ B & , \text{ falls } A = B . \end{cases}$$

Geben Sie das entsprechende, rekursiv arbeitende Unterprogramm an.

(Der Befehl XCHG reg1,reg2 (exchange) vertauscht die Inhalte der angegebenen Register. Die Parameterübergabe kann ohne Stapel erfolgen.)

3. Geben Sie für das unten angegebene Programm in Maschinensprache

- a) eine PASCAL-Version,
- b) mit Hilfe eines Schreibtischtests das Ergebnis (in DX),
- c) die mathematischen Zahlen, die das Programm allgemein berechnet, an.

```
HauptPro: MOV AX,5
          PUSH AX           ;Parameter1
          MOV AX,3
          PUSH AX           ;Parameter2
          CALL Gruebel
          RET

Gruebel:  MOV BP,SP
          MOV AX,[BP]+2
          CMP AX,1
          JZ Eins
          CMP AX,[BP]+4
          JZ Eins

          MOV AX,[BP]+4
          DEC AX
          PUSH AX
          MOV AX,[BP]+2
          DEC AX
          PUSH AX
          CALL Gruebel      ;Ergebnis1 in DX

          MOV BP,SP
          PUSH DX           ;Ergebnis1 auf Stapel
          MOV AX,[BP]+4
          DEC AX
          PUSH AX
          MOV AX,[BP]+2
          PUSH AX
          CALL Gruebel      ;Ergebnis2 in DX

          POP CX            ;Ergebnis1 vom Stapel
          ADD DX,CX
          RET 4

Eins:    MOV DX,1
          RET 4
```