

Klausur IA12.0/13.0/14.0/15.0 451 Assembler am 08.02.2005

Dauer : 120 Minuten

Anlage : Befehlssatz des 8088

Übersetze das nachfolgende TurboPascal-Programm *Zufaelle* in ein äquivalentes (d.h. möglichst bedeutungstreues) Assemblerprogramm (8086, EXE).

Program Zufaelle;

```
Type tpZahlen = ^tZahlen;
     tZahlen = Record
       Zahl      : Word;
       Naechste  : tpZahlen
     End;
```

```
Var I, Klein, Gross           : Word;
    Zufallszahl, Zufallszahlen : tpZahlen;
```

```
Procedure LuettGroot(Tohlen:tpZahlen;Var Luett,Groot:Word);
Begin
```

```
  Luett := 65535;
  Groot := 0;
  While Tohlen <> Nil Do Begin
    If Tohlen^.Zahl < Luett Then
      Luett := Tohlen^.Zahl;
    If Tohlen^.Zahl > Groot Then
      Groot := Tohlen^.Zahl;
    Tohlen := Tohlen^.Naechste
  End;
```

```
End;
```

```
Function Enthalten(Zahl:Word;Zahlen:tpZahlen):Boolean;
```

```
Begin
  If Zahlen = Nil Then
    Enthalten := False
  Else If Zahl = Zahlen^.Zahl Then
    Enthalten := True
  Else
    Enthalten := Enthalten(Zahl,Zahlen^.Naechste)
End;
```

```
End;
```

```
Begin
```

```
  Zufallszahlen := Nil;
  For I := 1 To 100 Do Begin
    New(Zufallszahl);
    Zufallszahl^.Zahl := Random(100)+1;
    Zufallszahl^.Naechste := Zufallszahlen;
    Zufallszahlen := Zufallszahl
  End;
  LuettGroot(Zufallszahlen,Klein,Gross);
  WriteLn('Am kleinsten ist die Zahl ',Klein,'.');
  WriteLn('Am groessten ist die Zahl ',Gross,'.');
  For I := 1 To 100 Do
    If Not Enthalten(I,Zufallszahlen) Then
      WriteLn('Die Zahl ',I,' ist nicht enthalten.');
```

```
  While Zufallszahlen <> Nil Do Begin
    Zufallszahl := Zufallszahlen;
    Zufallszahlen := Zufallszahlen^.Naechste;
    Dispose(Zufallszahl)
```

```
  End
```

```
End.
```

Tipps zur Bearbeitung der vorstehenden Aufgabenstellung :-)

Gemäß den Konventionen von Borland TurboPascal soll die Parameterübergabe (sowohl für Wert- als auch Referenzparameter) über den Stack und die Rückgabe eines Boolean-Funktionswerts über das AL-Register erfolgen. Dabei ist zu bedenken, das Referenzparameter aus einem 16-Bit-Segment- und 16-Bit-Offsetanteil bestehen.

Eine Pointer-Variable enthält die Adresse eines Wertes in der Reihenfolge Offset / Segment - Anteil. Bei der Übergabe einer Pointer-Variablen an ein Unterprogramm per Wertaufruf wird der Inhalt in der Reihenfolge Segment / Offset - Anteil auf den Stack gepusht.

Zur Realisierung der New-Prozedur und der Dispose-Prozedur werden die DOS-Funktionen "Allocate Memory" (Funktionscode 48h) und "Free Memory" (Funktionscode 49h) verwendet. Die Funktion zum Reservieren von Speicherplatz erwartet im BX-Register die Anzahl der benötigten zusammenhängenden 16-Byte Blöcke und liefert als Ergebnis im AX-Register die Segmentadresse der reservierten Blöcke (als Offsetadresse innerhalb des Segments wird Null impliziert). Der Fall "nicht genügend zur Verfügung stehender Speicher" bleibt unberücksichtigt. Die Funktion zum Freigeben von Speicherplatz erwartet im ES-Register die Segmentadresse der freizugebenden Blöcke, also eine Adresse die von der Funktion "Allocate Memory" zurückgeliefert wurde.

Im Modul TOOLS stehen die Routinen WWORD zur Ausgabe von ganzzahligen vorzeichenlosen Zahlen (der Datentransfer erfolgt über das DX-Register) und RANDOM zur Berechnung von ganzzahligen vorzeichenlosen "Zufallszahlen" im Intervall von Null bis "N-1" (der Datentransfer für "N" erfolgt über das DX-Register und für die "Zufallszahl" über das AX-Register) zur Verfügung.

Das PTL-Team wünscht viel Erfolg