Allgemeine Informatik II SoSe 2001

Prof. Dr. H. Neumann • Dr. K. Murmann • S. Geschwentner • Dr. F. Schwenker

3. Aufgabenblatt (bis zum 31.05.2001)

5. Aufgabe:

In dieser Aufgabe sollen einige Operationen implementiert werden, die den Umgang mit Zeitschriftenartikeln beschreiben. Dafür wird die folgende Datenstruktur verwendet :

```
CONST
   StringLength = 30;

TYPE
   string  = ARRAY [0..StringLength] OF CHAR;

   Article = RECORD
       title    : string;
       author   : string;
       pages    : CARDINAL;
       fontSize : CARDINAL;
     END;

   ArtPtr = POINTER TO Article;
```

Implementieren Sie die folgenden Prozeduren und Funktionsprozeduren.

1. `PROCEDURE MakeArticle(title, author : string; pages, fontSize : CARDINAL) : ArtPtr;`

   Diese Funktionsprozedur legt eine neue `Article`-Struktur an, füllt die Felder entsprechend der übergebenen Parameter ein, und gibt den Zeiger auf die Struktur zurück.

2. `PROCEDURE TrashArticle(VAR art : ArtPtr);`

   Diese Prozedur löscht den Artikel auf den `art` zeigt, indem sie den nicht mehr benötigten Speicherplatz zurückgibt, und setzt danach `art` auf `NIL`.

3. `PROCEDURE Matches(art : ArtPtr; word : string) : BOOLEAN;`

   Diese Funktionsprozedur liefert `TRUE` zurück, falls der Titel *oder* der Autorenname mit `word` übereinstimmt, andernfalls `FALSE`.

4. `PROCEDURE ReadingDuration(art : ArtPtr) : CARDINAL;`

   Diese Funktionsprozedur gibt die Zeit (in Minuten) zurück, die schätzungsweise gebraucht wird, um den Artikel `art` zu lesen. Diese berechnet sich wie folgt : pro Seite werden standardmässig 5 Minuten veranschlagt. Die sich daraus ergebende Gesamtdauer wird aber mit dem Fontgrössenfaktor $\frac{12}{\texttt{fontSize}}$ multipliziert, und auf die nächstgrösste ganze Zahl gerundet. Ausserdem soll der Ausnahmefall `fontSize`$= 0$ abgefangen werden; das Lesen beansprucht in diesem Fall keine Zeit.

5. `PROCEDURE ReadArticle(VAR art : ArtPtr);`

   Diese Prozedur liest eine Artikelbeschreibung von der Tastatur ein.

6. `PROCEDURE WriteArticle(art : ArtPtr);`

   Diese Prozedur gibt eine Artikelbeschreibung (auf dem Bildschirm) aus.

6. Aufgabe:
Entwickeln Sie Prozeduren und Funktionsprozeduren, die es Ihnen erlauben, Stapel von Artikeln (aus der Aufgabe 5) zu verwalten. Verwenden Sie dabei als Datenstruktur eine `linked list` (siehe Skript) mit den nachfolgenden Definitionen :

```
TYPE

  LINK = POINTER TO NODE;

  NODE = RECORD
           elem : ArtPtr;
           next  : LINK;
         END;

  STACK = RECORD
    head : LINK;
    z : LINK;
    height : CARDINAL;
  END;
```

Implementieren Sie die folgenden Prozeduren und Funktionsprozeduren. Passen Sie dazu die im Skript angegebene Beispiel-Implementierung an.

1. `PROCEDURE initStack(VAR s : STACK);`

   Diese Prozedur initialisiert den Stapel `s`.

2. `PROCEDURE stackEmpty(s : STACK) : BOOLEAN;`

   Diese Funktionsprozedur gibt `TRUE` zurück, falls `s` leer ist, andernfalls `FALSE`.

3. `PROCEDURE push(VAR s : STACK; art : ArtPtr);`

   Diese Prozedur legt den Artikel `art` zuoberst auf den Stapel `s`.

4. `PROCEDURE pop(VAR s : STACK) : ArtPtr;`

   Diese Funktionsprozedur nimmt den obersten Artikel vom Stapel `s`, und gibt ihn zurück. Falls `s` leer ist, gibt sie `NIL` zurück.

Schreiben Sie ein Programm, das zuerst einige Artikelbeschreibungen (mindestens 4; Titel, Autoren, etc selber ausdenken !) von der Tastatur einliest, und diese auf einen Stapel legt. Danach soll es die Artikel einen nach dem anderen vom Stapel nehmen, und ihre Beschreibung inklusive Lesedauer ausgeben.