**АНАЛИЗ НА РЕШЕНИЕТО НА ЗАДАЧА  
 КАЛЕНДАР**

След въвеждането на годината за която се отнася календара се пресмята броя на листите от които той ще е съставен. Към 365 се добавя страницата за Flip и евентуално 29 февруари (ако годината е високосна) и полученото число е броя на дните.

Функцията Init конструира календара в паметта, като за всеки лист се попълва следната структура:

typedef struct

{

int Page; // Номер на лист

int fd; // Ден

int fm; // Месец

int DayOfWeekFront; // Ден от седмицата на лицевата страница

int fds; // Ден ако е неделя

int fms; // Месец ако е неделя

int DayOfWeekFrontw; // Ден от седмицата на лицевата страница ако е неделя

int bd; // Ден гръб

int bm; // Месец гърб

int DayOfWeekBack; // Ден от седмицата на гърба

int bds; // Ден гръб, ако е неделя

int bms; // Месец гръб, ако е неделя

int DayOfWeekBackw; // Ден от седмицата на гърба ако е неделя//

} sheet ;

Дефинира се масив от структури:

sheet ART[CALENDARPAGES];

Дефинира се масив:

int DayIdx[YEARDAYS];

Този масив има толкова елементи, колкото са дните на годината и всеки елемент съдържа индекс към листа от календара (елемент на масива от структури ART), който съдържа информацията за този ден.

В един двоен цикъл: по месеци и вътрешен – по дни от съответния месец се конструира календара и се изгражда релацията <Пореден ден от годината> 🡪 <Лист от календара>.

След това е лесно. Въвежда се ден и месец. Пресмята се кой пореден ден от годината е тази дата. По индекс в DayIdx[<Ден от годината>] намираме съответния лист (елемент на масива от структури ART) 🡪 ART[DayIdx[<Ден от годината>]]. Разпечатва се информацията от структурата.

*Автори: Нели Манева, Пано Панов*