

```

; --- Úkol 3 ---
;
; Naprogramujte funkci 'task23', která alokuje a naplní pole 16bitových celých čísel bez znaménka Lucasovými čísly
; L(0), L(1), ... , L(N-1). Požadovaný počet Lucasových čísel je uveden v registru ECX (32bitová hodnota se znaménkem)
; a funkce vrací v EAX ukazatel na pole, které alokuje funkcí 'malloc' ze standardní knihovny jazyka C.
;
; Lucasova čísla jsou definována takto:
;
;   L(0) = 2
;   L(1) = 1
;   L(n) = L(n-1) + L(n-2)
;
; Vstup:
;   ECX = požadovaný počet prvků posloupnosti (32bitové celé číslo se znaménkem).
;
; Výstup:
;   EAX = 0, pokud N <= 0, nic nealokujete a vrátíte hodnotu 0 (NULL),
;   EAX = 0, pokud došlo k chybě při alokování paměti funkcí 'malloc' (vrátí hodnotu 0),
;   EAX = ukazatel na pole 16bitových celočíselných prvků bez znaménka reprezentujících Lucasova čísla.
;
; Důležité:
;   - funkce musí zachovat obsah všech registrů, kromě registru EAX a příznakového registru,
;   - funkce 'malloc' může změnit obsah registrů ECX a EDX.
;
; Funkce 'malloc' je definována takto:
;
;   void* malloc(size_t N)
;   N = počet bajtů, které mají být alokovány (32bitové celé číslo bez znaménka),
;   - funkce vrací v EAX ukazatel (32bitové celé číslo bez znaménka) na alokované místo v paměti,
;   - funkce vrací v EAX hodnotu 0 (NULL) v případě chyby při alokování paměti,
;   - funkce může změnit obsah registrů ECX a EDX.

```

```
; --- Úkol 3 ---
```

```
; Naprogramujte funkci 'task23', která alokuje a naplní pole 16bitových celých čísel bez znaménka Fibonacciho čísl  
; F(0), F(1), ... , F(N-1). Požadovaný počet Fibonacciho čísel je uveden v registru ECX (32bitová hodnota se znaménkem)  
; a funkce vrací v EAX ukazatel na pole, které alokuje funkcí 'malloc' ze standardní knihovny jazyka C.
```

```
; Fibonacciho čísla jsou definována takto:
```

```
; F(0) = 0  
; F(1) = 1  
; F(n) = F(n-1) + F(n-2)
```

```
; Vstup:  
; ECX = požadovaný počet prvků pole (32bitové celé číslo se znaménkem).
```

```
; Výstup:  
; EAX = 0, pokud N <= 0, nic nealokujete a vrátíte hodnotu 0 (NULL),  
; EAX = 0, pokud došlo k chybě při alokování paměti funkcí 'malloc' (vrátí hodnotu 0),  
; EAX = ukazatel na pole 16bitových celočíselných prvků bez znaménka reprezentujících Fibonacciho čísla.
```

```
; Důležité:  
; - funkce musí zachovat obsah všech registrů, kromě registru EAX a příznakového registru,  
; - funkce 'malloc' může změnit obsah registrů ECX a EDX.
```

```
; Funkce 'malloc' je definována takto:
```

```
; void* malloc(size_t N)  
; N = počet bytů, které mají být alokovány (32bitové celé číslo bez znaménka),  
; - funkce vrací v EAX ukazatel (32bitové celé číslo bez znaménka) na alokované místo v paměti,  
; - funkce vrací v EAX hodnotu 0 (NULL) v případě chyby při alokování paměti,  
; - funkce může změnit obsah registrů ECX a EDX.
```