

Termín odevzdání:	10.11.2023 16:59:59
Pozdní odevzdání s penalizací:	31.12.2023 23:59:59 (Penále za pozdní odevzdání: 100.0000 %)
Hodnocení:	0.4000
Max. hodnocení:	0.4000 (bez bonusů)
Odevzdaná řešení:	1 / 25
Nápovědy:	0 / 25

Úkolem je realizovat funkci (ne celý program, pouze funkci), která pro zadané datum rozhodne, kolikátý den v roce to je.

Funkce má následující rozhraní:

```
int dateToIndex (int day, int month, int year, int * idx);
```

- vstupními parametry funkce jsou den, měsíc a rok, které mají být ověřeny,
- `idx` je výstupní parametr udávající pořadové číslo zadaného dne od začátku roku. Pro 1. ledna je návratová hodnota 1. Pokud funkce uspěje, запиše pořadové číslo dne do svého výstupního parametru `idx`. Pokud funkce selže (neplatné hodnoty parametrů), ponechá výstupní parametr beze změn,
- návratovou hodnotou funkce je signalizace úspěchu (hodnota 1) a v případě chybných parametrů signalizace neúspěchu (hodnota 0).

Za chybu je považováno:

- `year` je menší než 2000 (všechna data před rokem 2000 považujeme za neplatná),
- `month` je mimo rozsah 1 až 12,
- `day` je neplatný (mimo 1 až počet dní v měsíci),

Ukázka použití funkce:

```
assert(dateToIndex( 1, 1, 2000, &days) == 1 && days == 1);
assert(dateToIndex( 1, 2, 2000, &days) == 1 && days == 32);
assert(dateToIndex(29, 2, 2000, &days) == 1 && days == 60);
assert(dateToIndex(29, 2, 2001, &days) == 0);
assert(dateToIndex( 1, 12, 2000, &days) == 1 && days == 336);
assert(dateToIndex(31, 12, 2000, &days) == 1 && days == 366);
assert(dateToIndex( 1, 1, 1999, &days) == 0);
assert(dateToIndex( 6, 7, 3600, &days) == 1 && days == 188);
assert(dateToIndex(29, 2, 3600, &days) == 1 && days == 60);
assert(dateToIndex(29, 2, 4000, &days) == 0);
```

Poznámky:

- Při výpočtu času uvažujeme Gregoriánský kalendář. Tedy měsíce mají vždy 30 nebo vždy 31 dní, výjimkou je únor, který má 28 dní (nepřestupný rok) nebo 29 dní (přestupný rok). Podle Gregoriánského kalendáře platí:
 - roky nejsou přestupné,
 - s výjimkou let dělitelných 4, které jsou přestupné,
 - s výjimkou let dělitelných 100, které nejsou přestupné,
 - s výjimkou let dělitelných 400, které jsou přestupné,
 - s výjimkou let dělitelných 4000, které nejsou přestupné.
- Tedy roky 2001, 2002, 2003, 2005, ... nejsou přestupné (pravidlo 1), roky 2004, 2008, ..., 2096, 2104, ... jsou přestupné (pravidlo 2), roky 2100, 2200, 2300, ... nejsou přestupné (pravidlo 3), roky 2000, 2400, ..., 3600, 4400, ... jsou přestupné (pravidlo 4) a roky 4000, 8000, ... nejsou přestupné (pravidlo 5).
- Jako základ Vašeho řešení použijte zdrojový kód z přiloženého souboru.
- Do funkce `main` si můžete doplnit i další Vaše testy, případně ji můžete libovolně změnit. Důležité je zachovat podmíněný překlad.
- V ukázce je použito makro `assert`. Pokud je parametrem nenulová hodnota, makro nedělá nic. Pokud je parametrem nepravda (nula), makro ukončí program a vypíše řádku, kde k selhání došlo. Pokud tedy Vaše implementace projde ukázkovými testy, program doběhne a nic nezobrazí.
- Update:** dostupné hlavičkové soubory jsou vidět v přiloženém vzoru. Jiné hlavičkové soubory nejsou k dispozici a nejsou ani potřeba. Nevkládejte další hlavičkové soubory pomocí `#include`, program skončí chybou kompilace.

Vzorová data:	Download
---------------	----------

☐ Referenční řešení

1	03.11.2023 11:27:57	Download
Stav odevzdání:	Ohodnoceno	
Hodnocení:	0.4000	

- Hodnotitel: automat

- Program zkompileován
- Test 'Zakladni test podle ukazky': Úspěch
 - Dosaženo: 100.00 %, požadováno: 100.00 %
 - Celková doba běhu: 0.000 s (limit: 1.000 s)
 - Úspěch v závazném testu, hodnocení: 100.00 %
- Test 'Test osetreni nespravnych vstupu': Úspěch
 - Dosaženo: 100.00 %, požadováno: 75.00 %
 - Celková doba běhu: 0.000 s (limit: 5.000 s)
 - Úspěch v nepovinném testu, hodnocení: 100.00 %
- Test 'Test nahodnymi daty': Úspěch
 - Dosaženo: 100.00 %, požadováno: 75.00 %
 - Celková doba běhu: 0.000 s (limit: 5.000 s)
 - Úspěch v nepovinném testu, hodnocení: 100.00 %
- Celkové hodnocení: 100.00 % (= 1.00 * 1.00 * 1.00)
- Celkové procentní hodnocení: 100.00 %
- Celkem bodů: $1.00 * 0.40 = 0.40$

SW metriky: