

Pracovní dny

Termín odevzdání:	27.11.2016 23:59:59
Hodnocení:	6.8750
Max. hodnocení:	5.0000 (bez bonusů)
Odevzdaná řešení:	2 / 20 Volné pokusy + 10 Penalizované pokusy (-10 % penalizace za každé odevzdání)
Nápovědy:	1 / 2 Volné nápovědy + 2 Penalizované nápovědy (-10 % penalizace za každou nápovědu)

Úkolem je realizovat dvě funkce (ne celý program, pouze dvě funkce), které budou usnadňovat výpočty pracovních dní v zadaném intervalu dat. Požadované funkce mají následující rozhraní:

```
int IsWorkDay ( int y, int m, int d )
    Funkce rozhodne, zda je zadaný den pracovní nebo ne. Parametry jsou rok, měsíc a den, který má být ověřen.
    Návrátovou hodnotou je celé číslo 1, pokud je zadaný den pracovní, nebo hodnota 0, pokud je zadaný den volný
    (sobota, neděle, svátek) nebo bylo zadané neplatné datum.

int CountWorkDays ( int y1, int m1, int d1, int y2, int m2, int d2, int * cnt )
    Funkce vypočte počet pracovních dní v zadaném intervalu. Interval je zadaný rokem, měsícem a dnem počátku
    (y1, m1, d1) a konce (y2, m2, d2). Interval považujeme za uzavřený, tedy do výpočtu zahrnujeme oba hraniční
    dny. Návrátovou hodnotou funkce je indikace úspěchu (hodnota 1) nebo neúspěchu (hodnota 0). Neúspěch bude
    funkce hlásit, pokud dostane neplatný časový interval (viz níže). Pokud funkce uspěje, umístí do výstupního
    parametru cnt počet nalezených pracovních dnů.
```

Správné hodnoty vstupních parametrů musí splňovat:

- rok je větší roven 2000 (všechna data před rokem 2000 považujeme za neplatná),
- měsíc je platný (1 až 12),
- den je platný (1 až počet dní v měsíci),
- ve funkci CountWorkDays je datum počátku intervalu \leq datum konce intervalu.

Odevzdávejte zdrojový soubor, který obsahuje implementaci požadovaných funkcí IsWorkDay a CountWorkDays. Do zdrojového souboru přidejte i další Vaše podpůrné funkce, které jsou z nich volané. Funkce budou volané z testovacího prostředí, je proto důležité přesně dodržet zadané rozhraní funkce. Za základ pro implementaci použijte kód z příloženého souboru. V kódu chybí vyplnit těla požadovaných funkcí (a případné další podpůrné funkce). Ukázka obsahuje testovací funkci main, uvedené hodnoty jsou použité při základním testu. Všimněte si, že vkládání hlavičkových souborů a funkce main jsou zabalené v bloku podmíněného překladu (#ifdef/#endif). Prosím, ponechte bloky podmíněného překladu i v odevzdávaném zdrojovém souboru. Podmíněný překlad Vám zjednoduší práci. Při kompilaci na Vašem počítači můžete program normálně spouštět a testovat. Při kompilaci na Progtestu funkce main a vkládání hlavičkových souborů "zmizí", tedy nebude kolidovat s hlavičkovými soubory a funkcí main testovacího prostředí.

Váš program bude spouštěn v omezeném testovacím prostředí. Je omezen dobou běhu (limit je vidět v logu referenčního řešení) a dále je omezena i velikost dostupné paměti. Časové limity jsou nastavené tak, aby rozumná implementace naivního algoritmu prošla všemi testy kromě testů bonusových, tedy byla hodnocena 100% bodů. Bonusové testy vyžadují časově efektivní výpočet i pro velké intervaly (vysoké roky hodně převyšující 4000).

Nápověda:

- Za pracovní dny tradičně považujeme pondělí až pátek. Dále za pracovní dny nepovažujeme státní svátky. Uvažujeme státní svátky podle následujícího přehledu:

1.1, 1.5, 8.5, 5.7, 6.7, 28.9, 28.10, 17.11, 24.12, 25.12 a 26.12

V seznamu chybí Velikonoce. Den Velikonoc je dán fází Měsíce, přesný výpočet je komplikovanější. V této úloze proto nebudeme Velikonoce uvažovat.

- 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:

1. roky nejsou přestupné,
2. s výjimkou let dělitelných 4, které jsou přestupné,
3. s výjimkou let dělitelných 100, které nejsou přestupné,
4. s výjimkou let dělitelných 400, které jsou přestupné,
5. 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.
- S trojicí hodnot (rok, měsíc, den) se špatně pracuje. Je lepší si tyto hodnoty převést na nějakou jinou reprezentaci, ideálně tak, aby vzniklo pouze jedno číslo.
- V programu musíte mnoho výpočtů dělat 2x - pro počátek a konec intervalu. Je dobrý nápad vytvořit si pomocné funkce, které 2x zavoláte.
- V povinných testech jsou zadávané roky nepřevyšující rok 2200.
- 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.

Vzorová data:

Download

Referenční řešení

2

19.11.2016 22:46:20

Download

Stav odevzdání: Ohodnoceno

Hodnocení: 6.8750

Hodnotitel: automat

- Program zkompileován
- Test 'Zakladni test podle ukazky': Úspěch
 - Dosaženo: 100.00 %, požadováno: 100.00 %
 - Celková doba běhu: 0.001 s (limit: 2.000 s)
 - Úspěch v závazném testu, hodnocení: 100.00 %
- Test 'Test mezních hodnot': Úspěch
 - Dosaženo: 100.00 %, požadováno: 20.00 %
 - Celková doba běhu: 0.002 s (limit: 10.000 s)
 - Úspěch v nepovinném testu, hodnocení: 100.00 %
- Test 'Test osetreni parametru': Úspěch
 - Dosaženo: 100.00 %, požadováno: 20.00 %
 - Celková doba běhu: 0.003 s (limit: 9.998 s)
 - Úspěch v nepovinném testu, hodnocení: 100.00 %
- Test 'Test nahodnymi daty': Úspěch
 - Dosaženo: 100.00 %, požadováno: 20.00 %
 - Celková doba běhu: 0.045 s (limit: 9.995 s)
 - Úspěch v nepovinném testu, hodnocení: 100.00 %
- Test 'Test nahodnymi daty (dlouhe intervaly, bonus #1)': Úspěch
 - Dosaženo: 100.00 %, požadováno: 100.00 %
 - Celková doba běhu: 3.027 s (limit: 10.000 s)
 - Úspěch v bonusovém testu, hodnocení: 125.00 %
- Test 'Test nahodnymi daty (dlouhe intervaly, bonus #2)': Program překročil přidělenou maximální dobu běhu
 - Vyčerpání limitu na celý test, program násilně ukončen po: 6.975 s (limit: 6.973 s)
 - Neúspěch v bonusovém testu, hodnocení: Bonus nebude udělen
- Celkové hodnocení: 125.00 % (= 1.00 * 1.00 * 1.00 * 1.00 * 1.25)
- Celkové procentní hodnocení: 125.00 %
- Bonus za včasné odevzdání: 0.50
- Celkem bodů: 1.25 * (5.00 + 0.50) = 6.88

		Celkem	Průměr	Maximum	Jméno funkce
SW metriky:	Funkce:	7	--	--	--
	Řádek kódu:	170	24.29 ± 23.69	81	main(...)
	Cyklomatická složitost:	121	17.29 ± 8.81	33	main(...)

1

19.11.2016 21:08:03

Download

Stav odevzdání: Ohodnoceno

Hodnocení: 0.0000

- **Hodnotitel: automat**

- ☐ Chyba při základní kompilaci
- Celkové procentní hodnocení: 0.00 %
- Bonus za včasné odevzdání: 0.50
- Celkem bodů: $0.00 * (5.00 + 0.50) = 0.00$