Seznam zaměstnanců II

Termín odevzdání: 30.12.2018 23:59:59

Pozdní odevzdání s penalizací: 06.01.2019 23:59:59 (Penále za pozdní odevzdání: 100.0000 %)

Hodnocení: 6.0000

Max. hodnocení: 5.0000 (bez bonusů)

Odevzdaná řešení:4 / 20 Volné pokusy + 10 Penalizované pokusy (-10 % penalizace za každé odevzdání)Nápovědy:0 / 2 Volné nápovědy + 2 Penalizované nápovědy (-10 % penalizace za každou nápovědu)

Úkolem je realizovat sadu funkcí pro práci se spojovými seznamy. Úřad je definován dvojicí spojových seznamů. První reprezentuje seznam pracovníků a druhý seznam automobilů. Pracovník je reprezentován jménem a může obsahovat odkaz na přidělený automobil. Automobil je reprezentován názvem modelu. S takovou strukturou chceme provádět tyto operace: inicializace úřadu, přidání pracovníka, přidání automobilu, kopírování úřadu a mazání úřadu.

TEMPLOYEE

tato datová struktura je deklarovaná v testovacím prostředí. Vaše implementace bude s touto strukturou pracovat, ale nesmí jí nijak měnit. Struktura reprezentuje jednoho zaměstnance. Zaměstnanci jsou uloženi v podobě jednosměrně zřetězeného spojového seznamu. Struktura má následující složky:

- m_Next odkaz na dalšího zaměstnance ve spojovém seznamu. Poslední zaměstnanec v seznamu má odkaz m Next nastaven na hodnotu NULL.
- m_Car odkaz na přidělený automobil. Odkaz má buď hodnotu NULL (pak automobil není přidělený), nebo odkazuje na nějaký prvek spojového seznamu automobilů.
- m Name ASCIIZ (nulou ukončený) řetězec udávající jméno zaměstnance.

TCAR

tato datová struktura je deklarovaná v testovacím prostředí. Vaše implementace bude s touto strukturou pracovat, ale nesmí jí nijak měnit. Struktura reprezentuje jeden automobil. Automobily jsou uložené v podobě jednosměrně zřetězeného spojového seznamu. Struktura má následující složky:

- m_Next odkaz na další automobil ve spojovém seznamu. Poslední automobil v seznamu má odkaz m_Next nastaven na hodnotu NULL.
- m Model ASCIIZ (nulou ukončený) řetězec udávající typ automobilu.

TOFFICE

tato datová struktura je deklarovaná v testovacím prostředí. Vaše implementace bude s touto strukturou pracovat, ale nesmí jí nijak měnit. Struktura reprezentuje jeden úřad a má následující složky:

- m Emp počátek spojového seznamu zaměstnanců,
- m_Car počátek spojového seznamu automobilů,

initOffice ()

funkce vytvoří nový prázdný úřad. Struktura úřadu bude dynamicky alokována, oba seznamy (zaměstnanci i automobily) budou prázdné.

addEmployee (office, name)

funkce vytvoří nový prvek ve spojovém seznamu zaměstnanců a umístí jej na první pozici v seznamu. Parametrem je name - jméno nového pracovníka a office - úřad, kam bude zaměstnanec přidán. Nově přidávaný zaměstnanec nemá přidělený automobil, tedy složka m Car bude mít hodnotu NULL.

addCar (office, model)

funkce vytvoří nový prvek ve spojovém seznamu automobilů a umístí jej na první pozici v seznamu. Parametrem je model - model nového automobilu a office - úřad, kam bude automobil přidán.

freeOffice (office)

funkce k uvolnění prostředků alokovaných ve spojových seznamech zadaného úřadu. Funkci je předán odkaz na úřad k uvolnění, funkce uvolní paměť používanou zaměstnanci i automobily zadaného úřadu a následně uvolní i strukturu vlastního úřadu.

cloneOffice (office)

funkce vytvoří kopii zadaného úřadu. Nově vytvořený úřad musí zachovat jména zaměstnanců i automobilů včetně jejich pořadí v seznamech a musí správně upravit odkazy na přidělené automobily. Pozor, nově vytvořený seznam musí být zcela nezávislý na originále, tedy i odkazy na automobily musí směřovat na prvky nově vytvořeného seznamu. Návratovou hodnotou funkce je odkaz na dynamicky alokovanou strukturu nového úřadu.

Odevzdávejte zdrojový soubor, který obsahuje implementaci požadovaných funkcí. Do zdrojového souboru přidejte i další Vaše podpůrné funkce, které jsou z nich volané. Implementované funkce budou volané z testovacího prostředí, je proto důležité

přesně dodržet zadané rozhraní funkcí. Za základ pro implementaci použijte kód z přiloženého 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. Rozumná implementace naivního algoritmu by měla projít všemi testy kromě testu rychlosti. Pro zvládnutí testu rychlosti je potřeba použít výkonnější algoritmus, který efektivně kopíruje seznamy.

Nápověda:

- 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.
- O jménech zaměstnanců a automobilů není obecně nic známo. Tedy jména mohou, ale nemusí být unikátní.

Vzorová data: _______Download

Referenční řešení

4	29.12.2018 20:41:12	Download
Stav odevzdání:	Ohodnoceno	
Hodnocení:	6.0000	

- Hodnotitel: automat
 - o Program zkompilován
 - Test 'Zakladni test podle ukazky': Úspěch
 - Dosaženo: 100.00 %, požadováno: 100.00 %
 - Celková doba běhu: 0.000 s (limit: 2.000 s)
 - Úspěch v závazném testu, hodnocení: 100.00 %
 - Test 'Test meznich hodnot': Úspěch
 - Dosaženo: 100.00 %, požadováno: 50.00 %
 - Celková doba běhu: 0.000 s (limit: 2.000 s)
 - Úspěch v závazném testu, hodnocení: 100.00 %
 - Test 'Test nahodnymi daty': Úspěch
 - Dosaženo: 100.00 %, požadováno: 50.00 %
 - Celková doba běhu: 0.011 s (limit: 2.000 s)
 - Úspěch v závazném testu, hodnocení: 100.00 %
 - Test 'Test nahodnymi daty + test prace s pameti': Úspěch
 - Dosaženo: 100.00 %, požadováno: 50.00 %
 - Celková doba běhu: 0.030 s (limit: 2.000 s)
 - Úspěch v nepovinném testu, hodnocení: 100.00 %
 - Test 'Bonus rychlost': Úspěch
 - Dosaženo: 100.00 %, požadováno: 100.00 %
 - Celková doba běhu: 2.453 s (limit: 5.000 s)
 - Úspěch v bonusovém testu, hodnocení: 120.00 %
 - Celkové hodnocení: 120.00 % (= 1.00 * 1.00 * 1.00 * 1.00 * 1.20)
- Celkové procentní hodnocení: 120.00 %
- Celkem bodů: 1.20 * 5.00 = 6.00

SW metriky:		Celkem	Průměr	Maximum Jméno funk	cce
	Funkce:	8	-		
	Řádek kódu:	257	32.13 ± 33.2	7 96 main	
	Cyklomatická složitost:	67	8.38 ± 11.89	38 main	

3 29.12.2018 10:59:21 Download

Stav odevzdání: Ohodnoceno Hodnocení: 0.0000

Hodnotitel: automat

Program zkompilován

Test 'Zakladni test podle ukazky': Úspěch

Dosaženo: 100.00 %, požadováno: 100.00 %
Celková doba běhu: 0.000 s (limit: 2.000 s)
Úspěch v závazném testu, hodnocení: 100.00 %

Test 'Test meznich hodnot': Úspěch

Dosaženo: 100.00 %, požadováno: 50.00 %
Celková doba běhu: 0.000 s (limit: 2.000 s)
Úspěch v závazném testu, hodnocení: 100.00 %

- Test 'Test nahodnymi daty': Program provedl neplatnou operaci a byl ukončen (Segmentation fault/Bus error/Memory limit exceeded/Stack limit exceeded)
 - Celková doba běhu: 0.006 s (limit: 2.000 s)
 - Neúspěch v závazném testu, hodnocení: 0.00 %
- Celkové hodnocení: 0.00 % (= 1.00 * 1.00 * 0.00)
- Celkové procentní hodnocení: 0.00 %
- Celkem bodů: 0.00 * 5.00 = 0.00

Celkem Průměr Maximum Jméno funkce

Cyklomatická složitost: **65 8.13** ± **11.96 38** main

2 28.12.2018 23:30:13 <u>Download</u>

Stav odevzdání:OhodnocenoHodnocení:0.0000

· Hodnotitel: automat

- Program zkompilován
- Test 'Zakladni test podle ukazky': Úspěch
 - Dosaženo: 100.00 %, požadováno: 100.00 %
 Celková doba běhu: 0.000 s (limit: 2.000 s)
 - Úspěch v závazném testu, hodnocení: 100.00 %
- Test 'Test meznich hodnot': Úspěch
 - Dosaženo: 100.00 %, požadováno: 50.00 %
 Celková doba běhu: 0.000 s (limit: 2.000 s)
 - Úspěch v závazném testu, hodnocení: 100.00 %
- Test 'Test nahodnymi daty': Program provedl neplatnou operaci a byl ukončen (Segmentation fault/Bus error/Memory limit exceeded/Stack limit exceeded)
 - Celková doba běhu: 0.006 s (limit: 2.000 s)
 - Neúspěch v závazném testu, hodnocení: 0.00 %
- Celkové hodnocení: 0.00 % (= 1.00 * 1.00 * 0.00)
- Celkové procentní hodnocení: 0.00 %
- Celkem bodů: 0.00 * 5.00 = 0.00

Celkem Průměr Maximum Jméno funkce

Funkce: 8 -- -- --**SW metriky:**Řádek kódu: **229 28.63 ± 32.14 96** main

Cyklomatická složitost: 61 7.63 ± 11.99 38 main

 1
 28.12.2018 21:34:23
 Download

 Stav odevzdání:
 Ohodnoceno

 Hodnocení:
 0.0000

· Hodnotitel: automat

- o Program zkompilován
- Test 'Zakladni test podle ukazky': Úspěch
 - Dosaženo: 100.00 %, požadováno: 100.00 %

Celková doba běhu: 0.000 s (limit: 2.000 s)
Úspěch v závazném testu, hodnocení: 100.00 %

Test 'Test meznich hodnot': Úspěch

Dosaženo: 100.00 %, požadováno: 50.00 %
Celková doba běhu: 0.000 s (limit: 2.000 s)
Úspěch v závazném testu, hodnocení: 100.00 %

- Test 'Test nahodnymi daty': Program provedl neplatnou operaci a byl ukončen (Segmentation fault/Bus error/Memory limit exceeded/Stack limit exceeded)
 - Celková doba běhu: 0.007 s (limit: 2.000 s)
 - Neúspěch v závazném testu, hodnocení: 0.00 %
- Celkové hodnocení: 0.00 % (= 1.00 * 1.00 * 0.00)

Celkové procentní hodnocení: 0.00 %

• Celkem bodů: 0.00 * 5.00 = 0.00

Celkem Průměr Maximum Jméno funkce

Cyklomatická složitost: **60 7.50 ± 11.95 38** main