

Dialogové okno I

Termín odevzdání:	06.05.2018 23:59:59
Pozdní odevzdání s penalizací:	01.07.2018 23:59:59 (Penále za pozdní odevzdání: 100.0000 %)
Hodnocení:	3.3000
Max. hodnocení:	3.0000 (bez bonusů)
Odevzdaná řešení:	5 / 20 Volné pokusy + 20 Penalizované pokusy (-2 % penalizace za každé odevzdání)
Nápovědy:	0 / 2 Volné nápovědy + 2 Penalizované nápovědy (-10 % penalizace za každou nápovědu)

Úkolem je navrhnout a implementovat sadu tříd, které budou simulovat prvky v uživatelském rozhraní. Konkrétně jde o okno (CWindow), tlačítko (CButton), textový vstup (CInput), textový popisek (CLabel) a combobox (CComboBox).

Úkol je zaměřen na návrh tříd, kde bude využito dědičnosti, polymorfismu a abstraktních metod. Pokud jsou tyto OOP prostředky použité racionálně, není implementace příliš dlouhá. Naopak, pokud provedete návrh špatně, bude se Vám kód opakovat a implementační soubor bude velký.

Navržené třídy budou GUI pouze simulovat. Nebudou nic vykreslovat, budou pouze vytvářet modely uživatelských prvků. Třídy tedy půjde použít k sestavení modelu uživatelských ovládacích prvků v okně, půjde manipulovat s jejich stavem a půjde zobrazit jejich stav ve výstupním proudu (viz níže). Jednotlivé uživatelské prvky půjde identifikovat pomocí jejich číselného id, pokud toto id je unikátní, pak pomocí tohoto id lze prvek i vyhledat. Pokud id unikátní není, bude ovládací prvek fungovat, ale nepůjde jej podle tohoto ID vyhledat.

Okno bude reagovat na změnu umístění/velikosti. Poloha a velikost jednotlivých ovládacích prvků je zadána relativně vzhledem k šířce/výšce nadřazeného okna (velikost/poloha je tedy desetinné číslo v rozsahu 0 až 1). Při změně velikosti okna se přepočtou pozice a velikosti jednotlivých ovládacích prvků. Při přepočtu se vynásobí odpovídající šířka (výška) desetinným číslem polohy/velikosti a pro zobrazení se použije takto vzniklé desetinné číslo.

Rozhraní tříd pro reprezentaci ovládacích prvků je velmi podobné. Konstruktor typicky má 2 nebo 3 parametry. Prvním parametrem bývá celočíselné id přiřazené ovládacímu prvku. Následuje informace o umístění (CRect se 4 složkami typu desetinné číslo, které udávají relativní pozici x, y a relativní velikosti w a h). Posledním parametrem bývá řetězec, který určuje zobrazovaný obsah. Dále, všechny třídy implementující ovládací prvky musí být schopné se zobrazit ve výstupním proudu operátorem << a musí být schopné se správně kopírovat (hluboké kopie). Testovací prostředí většinou požaduje pouze tuto základní funkcionalitu (tj. konstruktor, kopírování a operátor pro výstup). Vaše implementace ale zřejmě bude požadovat rozhraní další, které si do tříd budete muset doplnit.

CButton	Reprezentuje tlačítko. Testovací prostředí požaduje pouze konstruktor a operátor pro výstup.
CInput	Reprezentuje textový vstup. Posledním parametrem konstrukturu je řetězec, který je ve vstupním boxu zobrazován. Testovací prostředí požaduje konstruktor, operátor pro výstup a další dvě metody: <ul style="list-style-type: none"> <li>SetValue ( x ), která nastaví obsah na řetězec x,</li> <li>GetValue ( ), která vrátí aktuálně zobrazovaný řetězec.</li> </ul>
CLabel	Reprezentuje statický text. Posledním parametrem konstrukturu je zobrazovaný řetězec. Testovací prostředí požaduje konstruktor a operátor pro výstup.
CComboBox	Reprezentuje combobox. Testovací prostředí požaduje konstruktor, operátor pro výstup a další 3 metody: <ul style="list-style-type: none"> <li>Add ( x ), která přidá zadaný řetězec x do seznamu nabídek,</li> <li>SetSelected ( x ), která nastaví nabídku s indexem x jako vybranou (výchozí je vybraná nabídka s indexem 0),</li> <li>GetSelected ( ), která zjistí index aktuálně vybrané nabídky.</li> </ul>
CWindow	Reprezentuje okno. Třída má následující rozhraní: <ul style="list-style-type: none"> <li>konstruktor CWindow(title, absPos), který vytvoří okno s názvem title. Okno bude mít nastavenou pozici x,y a velikost w x h podle zadaného parametru absPos. Pozor, zde jsou rozměry absolutní.</li> </ul>

- Metoda `Add ( x )`, která přidá ovládací prvek `x`.
- Metoda `SetPosition ( absPos )`, která změní pozici/velikost okna. Zadané hodnoty jsou absolutní souřadnice a velikosti okna po změně. Metoda zároveň zajistí přepočítání pozice a velikosti všech ovládacích prvků obsažených v okně.
- Metoda `Search ( id )` vyhledá ovládací prvek podle zadaného `id`. Pokud prvek se zadaným `id` v okně neexistuje, metoda vrátí `NULL`. Pokud v okně existuje více ovládacích prvků se stejným `id`, metoda vrátí odkaz na některý z nich (např. na první).
- Operátor `<<` zobrazí obsah okna do výstupního proudu, formát je zřejmý z přiloženého zdrojového kódu.

CRect

Pomocná třída reprezentující obdélník. Tato třída je implementovaná v testovacím prostředí a je zahrnutá i do přiloženého zdrojového souboru. Při odevzdávání ji ponechte v bloku podmíněného překladu.

Odevzdávejte zdrojový kód s implementací tříd `CWindow`, `CButton`, `CInput`, `CLabel` a `CComboBox`. Do odevzdávaného souboru zahrňte všechny potřebné podpůrné deklarace. Části vkládání hlaviček a Vaše testy ponechte v bloku podmíněného překladu, jak je naznačeno v přiložené ukázce.

## Poznámky

- Používejte operátory pro přetypování (`dynamic_cast`) s rozmyslem. V ukázce byl použit, zde mu zcela zabránit nelze. Ale vlastní referenční implementace v sobě nemá žádné přetypování ani žádné použití RTTI. Obecně, RTTI, `dynamic_cast` a `typeid` vedou k více větvenému kódu, který je hůře čitelný a hůře rozšiřitelný. Navrhněte třídy tak, abyste si vystačili s polymorfismem.
- Vaše řešení musí používat třídy, třídy musí tvořit hierarchii, dědičnost a polymorfismus musí být použité. V této úloze je použití dědičnosti a polymorfismu vhodné, navíc, testovací prostředí odmítne řešení, které by dědičnost, polymorfismus a dynamicky vázané metody nevyužívalo (takové řešení bude odmítnuto na chybě při kompilaci).
- Výstup je ve formě "stromu", byť v této úloze je strom nejvýše dvojúrovňový. Číslo v hranatých závorkách udává `id` ovládacího prvku. V kulatých závorkách jsou pak absolutní souřadnice (`x,y,w,h`).
- Výpis obsahuje přepočtené absolutní souřadnice ovládacích prvků. Absolutní souřadnice jsou známy až po vložení ovládacího prvku do nějakého okna. Testovací prostředí proto nezkontroluje vytváření objektů ovládacích prvků před jejich vložení do nějakého okna.

Vzorová data:

Download

☐ Referenční řešení

5 28.04.2018 00:21:43

Download

Stav odevzdání: Ohodnoceno

Hodnocení: 3.3000

- **Hodnotitel: automat**
  - Program zkompileován
  - 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 navrhnu trid': 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 hodnotami': Úspěch
    - Dosaženo: 100.00 %, požadováno: 50.00 %
    - Celková doba běhu: 0.029 s (limit: 2.000 s)
    - Úspěch v závazném testu, hodnocení: 100.00 %
  - Test 'Test kopirujiciho konstruktora': Úspěch
    - Dosaženo: 100.00 %, požadováno: 50.00 %
    - Celková doba běhu: 0.001 s (limit: 1.971 s)
    - Úspěch v závazném testu, hodnocení: 100.00 %
  - Test 'Test operatoru=': Úspěch
    - Dosaženo: 100.00 %, požadováno: 50.00 %
    - Celková doba běhu: 0.001 s (limit: 1.970 s)
    - Úspěch v závazném testu, hodnocení: 100.00 %
  - Test 'Test nahodnymi daty + kontrola prace s pameti': Úspěch
    - Dosaženo: 100.00 %, požadováno: 50.00 %
    - Celková doba běhu: 0.011 s (limit: 2.000 s)

- Úspěch v závazném testu, hodnocení: 100.00 %
    - Celkové hodnocení: 100.00 % ( $= 1.00 * 1.00 * 1.00 * 1.00 * 1.00 * 1.00$ )
- Celkové procentní hodnocení: 100.00 %
- Bonus za včasné odevzdání: 0.30
- Celkem bodů:  $1.00 * (3.00 + 0.30) = 3.30$

		Celkem	Průměr	Maximum	Jméno funkce
<b>SW metriky:</b>	Funkce:	<b>17</b>	--	--	--
	Řádek kódu:	<b>89</b>	<b>5.24 ± 5.79</b>	<b>26</b>	Print(ostream &)
	Cyklomatická složitost:	<b>23</b>	<b>1.35 ± 1.88</b>	<b>8</b>	Print(ostream &)