

Objektovo orientované programovanie

2. zadanie

Vytvorte aplikáciu v ktorej budú aspoň dva typy ovládacích prvkov a aspoň dva typy zobrazovacích prvkov. Ovládací prvok bude používateľovi umožňovať meniť číselnú hodnotu. Na jeho výstup bude pripojený jeden alebo viac zobrazovacích prvkov, ktoré budú číselnú hodnotu zobrazovať. Typy ovládacích prvkov sa budú líšiť spôsobom ovládania, typy zobrazovacích prvkov spôsobom zobrazenia. Ovládacie prvky musia so zobrazovacími prvkami komunikovať pomocou sériového komunikačného protokolu, ktorý je definovaný rozhraním `ConnectableSerialInterface` uvedeným nižšie (zobrazovacie prvky budú implementovať rozhranie). Prepojenie medzi prvkami znázorníte aj vizuálne.

Program musí byť riadený udalosťami. Musí byť ovládaný myšou.

Pri implementácii dbajte na návrh a dobrú štruktúru programu. Vhodne použite princípy objektovo orientovaného programovania.

Ovládacie prvky budú mať zoznam referencií na zobrazovacie prvky, ktoré sú za ne zapojené. Pri zmene hodnoty oznámia pomocou protokolu novú hodnotu všetkým zapojeným zobrazovacím prvkom. Ovládacie prvky musia umožňovať pripojenie veľkého množstva zobrazovacích prvkov ľubovoľného typu implementujúceho zadané rozhranie (ak budete pri implementácii používať pole, môže byť počet obmedzený veľkosťou poľa).

Komunikácia bude vždy začínať volaním metódy `start()`, za ktorou nasleduje sériový prenos maximálne 7 bitov (metóda `nextData(boolean)`), ukončený volaním metódy `stop()`. Po ukončení komunikácie sa zobrazí nová hodnota. Využite aj konštanty definované v rozhraní.

Vytvorte triedy reprezentujúce ovládacie prvky a zobrazovacie prvky. Pre ovládacie prvky vytvorte spoločnú nadtriedu, do ktorej vhodne umiestnite spoločnú implementáciu. Ak to bude vhodné, vytvorte aj spoločnú nadtriedu pre zobrazovacie prvky. Triedy, ktorých inštancie nemá zmysel vytvárať označte ako abstraktné. Podľa potreby vytvorte aj ďalšie triedy.

Z dôvodu dobrého zapuzdrenia, definujte prístupové práva ku členom tried tak, aby boli prístupné len tam kde sú potrebné (`private`, `protected`, `public`). Tiež vhodne využívajte vnorené triedy.

Umiestnenie a prepojenie objektov môže byť v inicializačnej časti programu dané napevno, musí sa dať ale jednoducho zmeniť. Vytvorte viacero kombinácií zapojení.

Grafická reprezentácia stačí jednoduchá (schematická), ale musí byť prehľadná. Ovládanie programu musí byť intuitívne.

Na prezentáciu si pripravte zjednodušený diagram tried (na papieri, alebo elektronicky), obsahujúci názvy tried, vzťahy medzi triedami, prípadne ďalšie dôležité informácie.

Zadanie odovzdajte na 7. cvičení (5.11.) osobne a do AIS.

Definícia rozhraní:

//Suciastka, ktora ma jeden vstup. Pripojenie na tento vstup je mozne vizualne zobrazit

```
public interface Connectable {  
  
    //Vrati poziciu vstupneho portu suciastky  
    Position getInputConnectPosition();  
}
```

//Suciastka ktora prijima udaje cez seriove rozhranie

```
public interface ConnectableSerialInterface extends Connectable{  
  
    //minimalna hodnota ktoru moze prijat  
    int MIN_VALUE = 0;  
  
    //pocet bitov vstupnej informacie  
    int NUM_BITS = 7;  
  
    //maximalna hodnota ktoru mozno prijat (127)  
    int MAX_VALUE = (1 << NUM_BITS) - 1;  
  
    //zaciatok komunikacie  
    void start();  
  
    //koniec komunikacie  
    void stop();  
  
    //posielanie jedneho bitu (tato metoda sa vola max. NUM_BITS krat medzi volaniami start() a end())  
    void nextData(boolean bit);  
}
```