

Oblikovni obrasci u programiranju

završni ispit

1. Sučelja funkcija za čitanje sadržaja kazala deklarirana su kako slijedi.

```
// from dirent.h (POSIX.1-2001)
DIR *opendir(const char *name);
struct dirent *readdir(DIR *dir);
int closedir(DIR *dir);
```

Tipičan način korištenja tih funkcija prikazan je sljedećim primjerom:

```
void printDirectoryContents(char const* str){
    DIR *pDir = opendir(str);
    struct dirent *pEntry=0;
    while (pEntry = readdir(pDir)){
        printf("%s\n", pEntry->d_name);
    }
    closedir(pDir);
}
```

S kojim bi se oblikovnim obrascem mogao povezati prikazani primjer? Predloži sučelje razreda koji bi također pružao mogućnost pristupa nazivima datoteka zadanog kazala, ali u skladu s prepoznatim obrascem. Izradi izvedbu predloženog razreda korštenjem prikazanog proceduralnog sučelja.

2. Programski inženjeri su prilikom izrade korisničke aplikacije oblikovali sljedeće sučelje za čitanje sadržaja kazala:

```
class fs_base{
public:
    virtual void getDirectoryContents(
        char const* path, std::vector<std::string>& files) =0;
    //...
};

class fs_win32:public fs_base{
    // implementacija apstraktnih metoda
    // pod Windowsima
}
```

Naknadno je odlučeno da korisnička aplikacija treba raditi i pod POSIX-om, koristeći funkcije iz sučelja <dirent.h> prikazane u prvom zadatku.

Potrebno je isplanirati nadogradnju korisničke aplikacije bez mijenjanja postojećih razreda. Koji oblikovni obrazac je prikladno upotrijebiti u ovom slučaju? Predloži rješenje u obliku strukturnog dijagrama i odsječaka koda.

3. Kod izgradnje aplikacije za udaljeni rad s grozdovima računala oblikovani su konkretni razredi koji izvode zajedničko sučelje Cluster. U konkretnim razredima SGICluster i WindowsCluster dostupne su osnovne metode za pregledavanje trenutnog stanja svih računala u grozdu, pokretanja aplikacije na određenim računalima grozda i sl.

Za navedene računalne grozdove potrebno je oblikovati različite specifične operacije poput dinamičkog raspoređivanja poslova ili slanja poruka korisnicima, pri čemu se očekuje često dodavanje i izmjena takvih operacija. Potrebno je omogućiti transparentno pozivanje specifičnih operacija preko pokazivača na Cluster, ali bez dodavanja operacija u njegovo sučelje. Dakle, želimo da ishodište poziva ovisi i o konkretnom grozdu i o konkretnoj specifičnoj operaciji.

Potrebno je identificirati najprikladniji obrazac za rješenje problema, nacrtati odgovarajući dijagram razreda, te skicirati pripadajući izvorni kod.

4. U koji oblikovni obrazac bi se mogao uklopiti sljedeći razred? Opiši uloge razreda *X* i *Y* u okviru obrasca. Iz kojih komponenti se tipično poziva metoda *m1*? Nacrtaj strukturni dijagram obrasca, te skiciraj programske implementacije ostalih sudionika.

```
class Y: public X{
public:
    X* m1(){return new Y;}
    // ...
};
```

5. Razmatramo razvoj programa za upravljanje uređajem za naplatu debitnim karticama (tzv. POS terminal). Ispočetak je upravljački program pokretan izravno na terminalu, a u izvedbi je koristio sljedeću kolekciju metoda niske razine:

```
class POSControl{
public:
    POSTerminal();
    void configure(std::string cfg);
    void setBankCredentials(std::string request);
public:
    int countTransactions();
    std::string getTransaction(int index);
public:
    // ...
};
```

Međutim, kako je broj terminala rastao, javila se potreba da se upravljački program izvodi na odvojenom dedicanom računalu. Htjeli bismo postići da upravljački program bude neovisan o tome gdje se izvodi (na terminalu ili na udaljenom računalu). Koji bi nam obrazac mogao pomoći pri ostvarivanju tog cilja? Predloži rješenje, nacrtaj odgovarajući dijagram razreda, te skiciraj pripadajući izvorni kod.

6. Zadan je razred koji implementira dohvat, obradu i pohranu web stranica kako slijedi:

```
class WebPageProcessor{
public:
    void processUrl(std::string url);
private:
    std::string getPage(std::string url);
    std::string extractTextFromHTML(std::string html);
    void storeTextToDatabase(std::string text);
};

void WebPageProcessor::processUrl(std::string url){
    std::string html = getPage(url);
    std::string text = extractTextFromHTML(html);
    storeTextToDatabase(text);
}
```

Na početku razvoja bilo je važno što prije isporučiti verziju koja ispravno radi za protokol HTTP, stranice u HTML-u, te bazu MySQL. Međutim, sada je potrebno podržati i različite načine dohvata stranica (npr, FTP, lokalna datoteka, ...), dodatne formate podataka (npr, PDF, PS, ...), te dodatne baze podataka (npr, Oracle, Postgresql, ...).

Pokušaj poboljšati organizaciju koda kako bi se potrebna proširenja postigla u skladu s načelima oblikovanja. Da li rješenje problema odgovara nekom od oblikovnih obrazaca koji su predstavljeni na predavanjima?

7. U koji oblikovni obrazac bi se mogao uklopiti sljedeći razred? Opiši uloge razreda *X*, *Y* i *Z*. Nacrtaj dijagram razreda obrasca, te skiciraj programsku implementaciju metode *Z::method()*.

```
class X{
    virtual void method()=0;
    // ...
};
class Y: public X{
public:
    virtual void method();
    //...
};
```

```
class Z: public X{
public:
    virtual void method();
    //...
private:
    std::list<X*> m_;
    //...
};
```