

IJA:Zadání projektu

Zadání: Navrhněte a implementujte aplikaci pro návrh a editaci blokových schémat.

Poznámka: Zadání definuje podstatné vlastnosti aplikace, které musí být splněny. Předpokládá se, že detaily řešení si doplní řešitelské týmy.

Specifikace požadavků

1. Základní požadavky
 - aplikace umožní vytvářet, editovat, ukládat a načítat bloková schémata
 - každé schéma má svůj jedinečný název
 - vytvořená schémata lze uložit a opětovně načíst
 - schéma je složeno z bloků a propojů mezi bloky
2. Bloky
 - každý blok má definované vstupní a výstupní porty
 - s každým portem je spojen typ, který je reprezentován množinou dat v podobě dvojic název->hodnota; hodnota bude vždy typu double
 - bloky je možné spojit pouze mezi výstupním a vstupním portem
 - každý blok obsahuje výpočet (vzorce), které transformují hodnoty ze vstupních portů na hodnoty výstupních portů
3. Propojení mezi bloky
 - systém kontroluje kompatibilitu vstupního a výstupního portu propoje (stejný typ dat)
 - typ dat je přiřazen propoji automaticky podle spojených portů
4. Výpočet
 - po sestavení (načtení) schématu je možné provést výpočet
 - systém detekuje cykly v schématu; pokud jsou v schématu cykly, nelze provést výpočet
 - systém požádá o vyplnění dat vstupních portů, která nejsou napojena a poté postupně provádí výpočty jednotlivých bloků podle definovaných vzorců v každém bloku
 - při výpočtu se vždy zvýrazní blok, který je právě přepočítáván
 - výpočet lze krokovat (jeden krok = přepočet jednoho bloku)
5. Další podmínky
 - najetím myši nad propoj se zobrazí aktuální stav dat
 - zvažte způsob jednoduchého rozšiřování systému o nové bloky a data

Rozšíření pro tříčlenný tým

1. Je možné současně pracovat na více schématech.
2. Složené bloky
 - každé schéma může mít vstupní a výstupní porty a tím tvořit jeden složený blok
 - vytvořené schéma může být součástí jiného schématu jako složený blok; v takovém případě je zobrazen pouze jako blok s příslušnými vstupy a výstupy, lze zobrazit interní schéma tohoto složeného bloku
3. Realizujte způsob jednoduchého rozšiřování systému o nové bloky a data.

Součást odevzdání

- připravte předem alespoň 5 různých bloků a 3 různé typy (množiny) dat

Doporučení

- zamyslete se nad použitím vhodných návrhových vzorů

IJA: Pokyny k projektu

Podmínky vypracování projektu

- Pro realizaci projektu použijte Java SE 8.
- Pro grafické uživatelské rozhraní použijte JavaFX nebo JFC/Swing (<https://docs.oracle.com/javase/8/javase-clienttechnologies.htm>).
- Pro překlad a spuštění projektu bude použita aplikaci [ant](#).

- Projekt bude tvořen následující kořenovou adresářovou strukturou:

```
src/           - (adres.) zdrojové soubory (hierarchie balíků)
examples/     - (adres.) připravené datové soubory
build/        - (adres.) přeložené class soubory
doc/          - (adres.) vygenerovaná programová dokumentace
dest-client/  - (adres.) umístění výsledného jar archivu (+ dalších potřebných) po kompilaci klientské aplikace,
                příp. samostatné aplikace (pokud není řešena varianta klient-server),
                tento adresář bude představovat adresář spustitelné aplikace
dest-server/  - (adres.) umístění výsledného jar archivu (+ dalších potřebných) po kompilaci serverové aplikace (pokud je
                vyžadována při řešení klient-server varianty),
                tento adresář bude představovat adresář spustitelné aplikace
lib/          - (adres.) externí soubory a knihovny (balíky třetích stran, obrázky apod.), které vaše aplikace využívá
readme.txt    - (soubor) základní popis projektu (název, členové týmu, ...)
rozdeleni.txt - (soubor) soubor obsahuje rozdělení bodů mezi členy týmu (pokud tento soubor neexistuje, předpokládá se
                rovnoměrné rozdělení, vizte hodnocení projektu)
build.xml     - (soubor) build file pro aplikaci ant
```

- Všechny zdrojové soubory musí obsahovat na začátku dokumentační komentář se jmény autorů a popisem obsahu.
- Součástí projektu bude programová dokumentace. Dokumentace se vygeneruje při překladu aplikace a bude uložena v adresáři doc.
- Pokud vaše řešení vyžaduje další externí soubory (obrázky, jiné balíky apod.), umístěte je do adresáře lib.
- Přeložení aplikace:
 - v kořenovém adresáři se provede příkazem [ant compile](#)
 - zkompilují se zdrojové texty, class soubory budou umístěny v adresáři build
 - vytvoří se programová dokumentace a uloží se do adresáře doc
 - vytvoří se jar archivy s názvy `ija-client.jar` v adresáři `dest-client` a `ija-server.jar` v adresáři `dest-server` (pokud je vyžadováno); do těchto adresářů se vytvoří/nakopírují další potřebné soubory a archivy
- Spuštění aplikace:
 - spouštět se budou vytvořené archivy jar
 - v kořenovém adresáři se provede příkazem [ant run](#)
 - spustí se klientská aplikace, případně server a dvě klientské aplikace (podle realizované varianty)

Podmínky odevzdání projektu

- Adresářovou strukturu umístěte do kořenového adresáře s názvem, který odpovídá loginu vedoucího týmu, např. `xloginXX`. Pro obsah podadresářů platí:
 - Adresáře `build`, `doc`, a `dest-*` budou prázdné.
 - V adresáři `examples` budou ukázkové datové soubory (pokud je zadání vyžaduje).
 - Adresář `lib`
 - obsahuje pouze skript s názvem `get-libs.sh`, který po spuštění stáhne z internetu požadované externí knihovny či soubory (předpokládejte prostředí Unix/Linux a přítomnost nástroje `wget` (v. 1.15) a `zip` (v.3.0))
 - externí knihovny lze stáhnout přímo ze zdroje, další soubory můžete dočasně umístit na váš webový prostor
 - pokud vaše řešení žádné další knihovny a soubory nepožaduje, nedělá skript nic
- Obsah kořenového adresáře zabalte do archivu zip:
 - Název archivu bude stejný jako název kořenového adresáře (s příponou `.zip`), tj. `login vedoucího týmu` (např. `xloginXX.zip`). Po rozbalení archivu vznikne adresářová struktura definovaná výše včetně adresáře `xloginXX`.
 - Archiv zip odevzdá pouze *vedoucí týmu* do informačního systému, termín *Projekt*.

Hodnocení projektu

- Projekt je hodnocen na stupnici 0-100. Po ohodnocení dojde k přepočtu na body pro jednotlivé členy týmu, a to rovnoměrně (každý získá stejný počet bodů, který odpovídá hodnocení projektu). Hodnocení každého člena týmu lze přerozdělit:
 - přerozdělení bodů je možné v rozsahu -50% až +50%
 - při přerozdělení může být vzata v úvahu aktivita jednotlivých členů
 - pokud máte dojem, že některý člen týmu nepracuje a nezaslouží si ani 50% z hodnocení, je nutno toto řešit individuálně s dr. Kočím **před odevzdáním projektu**
 - přerozdělení bodů musí být zapsáno v souboru `rozdeleni.txt` v následujícím formátu (uvádí se procenta, která získává člen týmu; součet musí odpovídat hodnotě `pocet_clenu * 100`):

```
xloginXX: 50
xloginYY: 150
```

- Hodnocení zahrnuje kvalitu OO návrhu, kvalitu implementace, dodržování stylu psaní programu (odsazování, kvalita komentářů, vhodné identifikátory) a především funkčnost programu.
- Při návrhu postupujte tak, aby výsledný program byl dobře použitelný. Hodnotit se bude nejen to, zda splňuje kladené požadavky, ale také JAK je splňuje.

- Pro získání zápočtu musí mít každý člen týmu (po příp. korekcích v rámci týmu) alespoň 50% z maxima hodnocení projektu.
- **Doporučuji po vytvoření archivu přenést a vyzkoušet přeložitelnost a spuštění aplikace v jiném prostředí (např. merlin.fit.vutbr.cz).**
- **Pokud nepůjde projekt přeložit, bude hodnocen 0 body!**

IJA:Prezentace projektů

Registrace týmu na prezentaci

- Tým si vybere ve WIS vhodný termín (vizte varianty termínu *Prezentace projektu*) a vedoucí týmu se na tento termín zaregistruje. Místnost, ve které bude prezentace probíhat, bude uvedena u příslušné varianty.
- Registruje se pouze vedoucí týmu; vedoucí se registruje vždy, i když se nemůže prezentace účastnit.

Postup prezentace

- Přineste s sebou notebook s připraveným **spuštěným projektem pro demonstraci**; pokud nemáte možnost přijít s notebookem, bude na místě k dispozici notebook s odevzdaným řešením.
- Připravte si základní demonstraci vašeho projektu podle vybrané varianty (vizte níže); asi na 4 - 5 minut
- Po demonstraci zlehka prodiskutujeme váš návrh (koncept, rozdělení do balíků, zda jste aplikovali některý architektonický vzor apod.)
- Při demonstraci
 - **se soustředíte pouze na demonstraci požadovaných vlastností** - nevysvětľujte, proč jste zvolili takové rozmístění grafických komponent apod., tyto otázky se dají rozvinout v případné diskuzi
 - **neukazujte překlad a spuštění aplikace** - to bude ověřeno před prezentací
- Hodnocení
 - Body za projekt budou přiděleny na základě splnění formálních požadavků a prezentace. Body budou přiděleny ihned po prezentaci.
 - Ukončením prezentace a přidělením bodů se uzavírá hodnocení týmu.
 - Je vhodné, aby prezentaci projektu byli přítomni všichni členové týmu; lze pověřit jen vybrané členy, v tom případě ostatní členové přijímají výsledek bez výhrad.

Demonstrace základní varianty

1. vytvoření jednoduchého schématu (pár propojených bloků), ukázka provedení výpočtu včetně zadání potřebných dat na vstupu
2. uložení a opětovné načtení schématu
3. načtení schématu s cykly, ukázka detekce cyklu
4. načtení schématu, ukázka krokování výpočtu
5. ukázka zobrazení dat po najetí myši nad propoj

Demonstrace rozšířené varianty pro tříčlenný tým

1. vytvoření jednoduchého schématu (pár propojených bloků), ukázka provedení výpočtu včetně zadání potřebných dat na vstupu
2. uložení a opětovné načtení schématu
3. načtení schématu s cykly do nového okna/tabu, ukázka detekce cyklu
4. načtení schématu do nového okna, ukázka krokování výpočtu
5. ukázka zobrazení dat po najetí myši nad propoj
6. načtení schématu, které obsahuje jiné schéma jako složený blok; ukázka editace schématu a výpočtu
7. demonstrace rozšiřování aplikace o nové bloky a schémata