**Dokumentace k aplikaci pro správu skladu**

**1. Formulace zadání**

Cílem této aplikace je vytvořit systém pro správu skladu, který umožní uživatelům přidávat, odstraňovat, vyhledávat a zobrazovat zboží. Aplikace by měla také poskytovat možnost ukládat data do souboru a načítat je zpět do programu. Důraz byl kladen na ošetření špatných vstupů, aby se minimalizovaly chyby způsobené uživatelským zadáním.

**2. Analýza problému**

V rámci správy skladu je nutné zajistit efektivní manipulaci se zbožím, což zahrnuje:

* Přidání nového zboží do skladu
* Odebrání zboží ze skladu
* Vyhledání zboží podle jeho ID
* Zobrazení všech položek ve skladu
* Vypočítání celkové ceny zboží
* Uložení a načtení dat do/z files
* Ošetření špatných vstupů pro zajištění správného fungování aplikace
* **Řazení položek podle různých kritérií** (např. podle ceny nebo názvu)

**3. Odůvodnění zvolených algoritmů**

* **Mapování (std::map)**: Zajišťuje rychlý přístup a úpravy položek pomocí ID jako klíče. To znamená, že pro přidání, odstranění nebo vyhledání položky je potřeba průměrně O(log n) času, což je efektivní i pro větší množství zboží.
* **I/O operace**: Použití standardních knihoven pro čtení a zápis do souborů, což je vhodné pro jednoduché manipulace s textovými daty. Algoritmy pro čtení a zápis zboží do souboru se opírají o sekvenční přístup, kde se každá položka zpracovává po jednom, což usnadňuje zpracování a identifikaci chyb.

**4. Popis realizace**

Aplikace je rozdělena do několika tříd:

* **Zbozi**: Třída reprezentující položku skladu, která obsahuje atributy jako ID, název, cena a počet kusů.
* **Sklad**: Třída pro správu kolekce položek a implementaci funkcí pro přidání, odstranění, vyhledání a uložení zboží.
* **main**: Hlavní funkce, která poskytuje uživatelské rozhraní pro interakci se skladovým systémem a ošetřuje uživatelské vstupy.

**5. Postup testování**

Aplikace byla testována na základě různých scénářů, včetně:

* Přidání položek do skladu: Testovali jsme, zda se položky správně ukládají a zda se jejich počet zvyšuje podle očekávání.
* Odebrání položek: Ověřovali jsme, že položky byly skutečně odstraněny a že se již nezobrazují v seznamu.
* Vyhledávání existujících a neexistujících položek: Zjistili jsme, zda vyhledávací funkce správně funguje jak pro existující, tak pro neexistující ID.
* Kontrola správnosti celkových cen a správného uložení do souboru: Testovali jsme, zda aplikace správně vypočítává celkovou cenu a zda se data správně ukládají do souborů.
* **Ošetření špatných vstupů**: Testovali jsme, jak aplikace reaguje na neplatné vstupy, například text místo čísel.

**6. Popis algoritmů**

1. **pridatZbozi()**
   * **Popis**: Metoda přidá novou položku do skladu.
   * **Jak funguje**: Aplikace přečte vstupní údaje o zboží (ID, název, cena, počet kusů) z konzole. Následně vytvoří objekt třídy Zbozi a přidá ho do mapy (std::map), kde je ID zboží použito jako klíč a samotný objekt zboží jako hodnota. Pokud existuje zboží se stejným ID, je přepsáno novými údaji. Ošetřuje špatné vstupy.
2. **odebratZbozi()**
   * **Popis**: Metoda odebere zboží na základě jeho ID.
   * **Jak funguje**: Uživateli je vyžádáno zadání ID zboží, které chce odstranit. Program následně zkontroluje, zda se dané ID nachází v mapě. Pokud ano, pomocí funkce erase() se položka odstraní. Pokud položka neexistuje, nedojde k žádné změně. Ošetřuje špatné vstupy.
3. **vyhledatZbozi()**
   * **Popis**: Metoda najde zboží podle ID a vypíše jeho detaily.
   * **Jak funguje**: Uživatel zadá ID zboží. Metoda vyhledá toto ID v mapě pomocí funkce find(). Pokud je položka nalezena, vrátí její ukazatel, jinak vrátí nullptr. Při úspěšném nalezení jsou detaily zboží vypsány na obrazovku. Ošetřuje špatné vstupy.
4. **vypsatVsechnoZbozi()**
   * **Popis**: Metoda vypíše všechny položky skladu.
   * **Jak funguje**: Aplikace prochází celou mapu obsahující zboží pomocí smyčky for. Pro každou položku zavolá metodu vypisZbozi(), která vypíše detaily zboží na obrazovku. Pokud je sklad prázdný, zobrazí se zpráva "Sklad je prázdný."
5. **vypocetCelkoveCeny()**
   * **Popis**: Metoda vypočítá celkovou cenu všech položek skladu.
   * **Jak funguje**: Aplikace prochází mapu obsahující zboží a pro každou položku volá metodu celkovaCena(), která vynásobí cenu položky počtem kusů. Tyto hodnoty se sečtou a celková cena je zobrazena uživateli.
6. **ulozitDoSouboru()**
   * **Popis**: Metoda uloží všechna data o zboží do souboru ve formátu CSV.
   * **Jak funguje**: Aplikace otevře výstupní soubor, projde každou položku v mapě a zapíše její ID, název, cenu a počet kusů do souboru, oddělené čárkami. Po dokončení se soubor uzavře.
7. **nacistZeSouboru()**
   * **Popis**: Metoda načte zboží ze souboru a vloží ho zpět do skladu.
   * **Jak funguje**: Aplikace otevře zadaný soubor a čte ho po řádcích. Každý řádek obsahuje informace o zboží ve formátu CSV (ID, název, cena, počet kusů). Tyto informace jsou parsovány a přidány do mapy skladu. Pokud dojde k chybě při čtení souboru, zobrazí se chybová zpráva.
8. **seradZbozi()**
   * **Popis**: Metoda seřadí zboží podle zvoleného kritéria (např. podle ceny nebo názvu).
   * **Jak funguje**: Aplikace uživateli nabídne možnost vybrat kritérium pro řazení (např. cena nebo název). Následně aplikace seřadí položky ve skladu podle vybraného kritéria a vypíše je na obrazovku.

**7. Manuál k ovládání aplikace**

**Spuštění aplikace**

1. Kompilujte kód v C++ a spusťte program.
2. V menu se zobrazí následující možnosti:
   * **1. Přidat zboží**: Umožňuje uživateli přidat nové zboží do skladu. Po zadání ID, názvu, ceny a počtu kusů se zboží přidá do interní struktury aplikace. Ošetřuje neplatné vstupy.
   * **2. Odebrat zboží**: Umožňuje uživateli odstranit zboží ze skladu na základě jeho ID. Pokud je položka nalezena, bude odstraněna. Ošetřuje neplatné vstupy.
   * **3. Vyhledat zboží**: Umožňuje uživateli vyhledat zboží podle ID. Pokud je položka nalezena, zobrazí se její podrobnosti; jinak se zobrazí zpráva o nenalezení zboží. Ošetřuje neplatné vstupy.
   * **4. Vypsat všechno zboží**: Zobrazí všechny položky ve skladu, včetně jejich ID, názvu, ceny a počtu kusů. Zobrazí zprávu "Sklad je prázdný", pokud žádné položky nejsou.
   * **5. Vypočítat celkovou cenu zboží**: Vypočítá a zobrazí celkovou cenu všech položek ve skladu. To zahrnuje součet cen všech položek, přičemž se bere v úvahu jejich počet.
   * **6. Uložit do souboru**: Umožňuje uživateli uložit aktuální stav skladu do textového souboru. Uživatel zadá název souboru a data budou uložena ve formátu CSV.
   * **7. Načíst ze souboru**: Umožňuje uživateli načíst zboží ze souboru. Uživatel zadá název souboru a data budou načtena do aplikace. Pokud se vyskytnou chyby, aplikace to oznámí.
   * **8. Řadit zboží**: Umožňuje uživateli seřadit zboží podle vybraného kritéria. Aplikace se ptá na kritérium (např. cena nebo název) a zobrazí se seřazený seznam.
   * **0. Konec**: Ukončení programu. Všechny otevřené soubory se automaticky zavřou a aplikace se ukončí.

**8. Příklady použití**

* Pro přidání zboží zvolte možnost 1 a zadejte požadované údaje. Ověřte, že se program ptá na platnost vstupů (např. čísla).
* Chcete-li zobrazit všechny položky, vyberte možnost 4. Ověřte, že se zobrazí zpráva "Sklad je prázdný", pokud žádné položky nejsou.
* Pro seřazení zboží zvolte možnost 8 a vyberte kritérium pro řazení.