

Warsztat programisty – Zadanie projektowe.

Etap 1

Łukasz Kuszner, Radosław Ziemann

14 listopada 2025

W tym etapie zakładamy, że dane wejściowe odpowiadają specyfikacji podanej w opisie poniżej. Zadanie polega na kontrolowaniu stanu magazynu i tworzeniu prostych raportów. Kluczowy element to prawidłowa konfiguracja projektu.

Wymagania formalne

Projekt składa się z:

- przynajmniej jednego pliku konfiguracyjnego dla systemu `CMake`,
- przynajmniej jednego pliku źródłowego w `C`,
- przynajmniej jednego skryptu w `Bashu`

Działanie programu

Program obsługuje następujące komendy:

1. `init` – pierwsze użycie i utworzenie magazynu
2. `update` – aktualizacja stanu magazynu
3. `print` – wypisanie aktualnego stanu magazynu do pliku w formacie `.txt` lub `.pdf` w formie tabeli zawierającej wszystkie przechowywane towary wraz z kodem, nazwą i aktualnym stanem w magazynie.

Format danych wejściowych

Proszę przyjąć, że dane są w postaci tekstu (ang. plain text) kodowanego w ASCII. W danych występują znaki z zakresu podstawowego (0 – 127).

Format danych dla komendy init

W pierwszej linii znajduje się jedna liczba całkowita n z przedziału $[1..1000]$, oznaczająca liczbę możliwych towarów przechowywanych w magazynie. W kolejnych n liniach podane są **kod towaru** i **nazwa towaru**.

Kod towaru to napis złożony z 5 znaków, z których pierwsze dwa to duże litery alfabetu łacińskiego, a kolejne trzy to cyfry.

Nazwa towaru to napis złożony z małych i dużych liter alfabetu łacińskiego, cyfr i spacji zaczynający się od litery.

Przykład

```
3
AB234 R12
SD121 R20
RS001 Zestaw baterii
```

W powyższym przykładzie mamy magazyn utworzony dla trzech możliwych towarów.

Format danych dla komendy update

W pierwszej linii jedna liczba całkowita n z przedziału $[1..100]$ oznaczająca liczbę aktualizacji do wykonania. W kolejnych n liniach podane są: **kod towaru** znak $+$ lub $-$ i liczba naturalna z przedziału $[1..2000]$ oznaczająca odpowiednio zwiększenie lub zmniejszenie liczby przechowywanego towaru.

Przykład

```
2
AB234 + 8
RS001 - 3
```

W powyższym przykładzie dodano 8 sztuk towaru AB234 i odjęto 3 sztuki towaru RS001.

Format danych dla komendy print

Komenda print nie wymaga żadnych danych, a jedynie dodatkowego parametru – podstawowej nazwy pliku, do którego ma być zapisany wynik komendy. Nazwa pliku powinna składać się z 10 znaków: małych liter alfabetu łacińskiego i cyfr zaczynających się od litery.

Scenariusze użycia

Scenariusz użycia składa się z sekwencji komend. Pierwszą komendą jest **init** a następnie występuje pewna liczba komend **update** i **print**.

Na potrzeby zadania zakładamy, że opis scenariusza znajduje się w pliku, w którym komendy `init` i `update` podane są wraz z plikami, w których znajdują się dane dla tych komend.

Przykład

Przypuśćmy, że plik `i.txt` zawiera:

```
3
AB234 R12
SD121 R20
RS001 Zestaw baterii
```

a plik `u.txt` zawiera

```
2
AB234 + 8
RS001 + 2
```

Wtedy program uruchomiony w trybie `txt` ze scenariuszem:

```
init i.txt
print report1211
update u.txt
update u.txt
print report1411
```

powinien wygenerować pusty plik `report1211.txt` oraz plik `report1411.txt` zawierający:

```
AB234 16 R12
RS001 4 Zestaw baterii
```

albo (kolejność nie jest narzucona):

```
RS001 4 Zestaw baterii
AB234 16 R12
```

Spodziewane efekty

- Projekt jest poprawnie skonfigurowany przy pomocy `cmake`.
- Historia zmian w repozytorium `git` jest niepusta i pokazuje kolejne etapy tworzenia projektu.
- Program można uruchomić w dwóch trybach: `pdf` i `txt`. Tryb uruchomienia podany jest jako parametr wywołania. Od trybu uruchomienia zależy format plików wynikowych.

- W trybie `txt`, dla każdej komendy `print` program generuje plik z nazwą taką jak podano w scenariuszu uruchomienia i doklejonym rozszerzeniem `txt`. W wygenerowanym pliku znajduje się opis stanu magazynu. Dla każdego towaru, który ma stan większy niż zero generowany jest wiersz z kodem towaru liczbą sztuk w magazynie i jego nazwą.
- W trybie `pdf` podobnie, tylko plik ma doklejone rozszerzenie `pdf` a w pliku znajduje się tabela z opisem stanu magazynu. W każdym wierszu tabeli, w pierwszej kolumnie znajduje się kod towaru, w drugiej liczba sztuk a w trzeciej nazwa.
- Stan magazynu przechowywany jest w pliku i po przetworzeniu każdej komendy jest on aktualizowany.
- W przypadku podania poprawnego scenariusza uruchomienia i dostępności założonych zasobów, powstają poprawne pliki `.pdf` lub `.txt` dla każdej komendy `print`, a każde wywołanie programu kończy się kodem 0 (wykonanie bezbłędne).