

Kodér/dekodér čárového kódu EAN-13

Zadání

Vytvořte software pro kódování a dekódování čárového kódu EAN-13. Vstupem pro kódování bude 12místné číslo. Kodér spočítá 13. kontrolní číslici. Vstupem pro dekódování bude 84místné binární číslo. Dekodér převede binární kód zpět do číselné podoby a detekuje při tom chyby.

Teoretický rozbor

Čárový kód EAN-13

Čárový kód EAN-13 je numerický kód pevné délky. Obsahuje dva stejné okrajové znaky (start/stop), dělicí znak a 13 numerických znaků (číslíce 0 až 9). Nosičem informace v kódu EAN-13 jsou čáry i mezery. Mohou mít různou šířku, která je ale vždy násobkem šířky nejužšího elementu. Šířka nejužšího elementu představuje základní délkovou jednotku. Každý zakódovaný znak má v symbolu čárového kódu EAN-13 délku přesně 7 základních jednotek.

Na základě pevné délky kódu znaku (7 jednotek) byly vytvořeny pro kódování číslic 0 až 9 tři znakové sady, pojmenované A, B a C. Symbol čárového kódu EAN-13 je uprostřed rozdělen dělicím znakem na dvě části, zvané pole. První pole je vlevo od dělicího znaku, druhé pole je vpravo.

Pro kódování prvního pole se nepoužívá jen jedna znaková sada, ale použijí se dvě. Dělá se to pro to, že do prvního pole je třeba umístit 7 znaků (kód země a kód výrobce). První číslice kódu země není přímo reprezentována čárovým kódem, ale vyplývá z označení sad, z níž se berou kódy pro 2. až 7. číslici, tj. šestici číslic v prvním poli.

V druhém poli je zakódováno číslo výrobku (5 číslic) a znak kontrolního součtu (1 číslice). Všechny znaky jsou zde kódovány ze znakové sady C.

Znak kontrolního součtu vypočteme tak, že všech 12 číslic podrobíme následující matematické operaci.

- 1) Sečtou se hodnoty číslic na sudých pozicích a výsledek se vynásobí třemi.
- 2) Sečtou se hodnoty číslic na lichých znakových pozicích.
- 3) Sečtou se obě výsledné hodnoty a výsledek se zaokrouhlí nahoru na celé desítky.
- 4) Kontrolní číslice je rozdíl zaokrouhlené a původní hodnoty.

Kódovací tabulka EAN			
znak	sada A	sada B	sada C
0			
1			
2			
3			
4			
5			
6			
7			
8			
9			
okrajové znaky		dělicí znak	

Dekódování první číslice

Číslice	Znakové sady	Číslice	Znakové sady
0	AAAAAA	5	ABBAAB
1	AABABB	6	ABBBAA
2	AABBAB	7	ABABAB
3	AABBBA	8	ABABBA
4	ABAABB	9	ABBABA

Kodér

Ze vstupních 12ti číslic spočte kontrolní číslici, podle výše popsaných kroků. Podle první číslice kódu zjistíme, jaká kombinace sad pro prvních 6 znaků je použita. Následně provede převod 12 znaků, bez první číslice, do binární podoby, která se použije pro grafické vykreslení čárového kódu.

Dekodér

Vstupní řetězec binárních čísel rozdělí po 7 znacích. Pro prvních 6 takto vzniklých řetězců ověří jejich existenci v sadách A a B. Pokud daný řetězec není ani v jedné z těchto sad, dekodér detekuje chybu. Takto zjistíme posloupnost použitých sad pro prvních šest znaků. Z této kombinace dekodér určí první číslici čárového kódu. Pokud tato kombinace neodpovídá žádnému znaku, je opět detekována chyba. Zbýlých 6 řetězců je porovnáváno s kombinacemi v sadě C.

Implementace

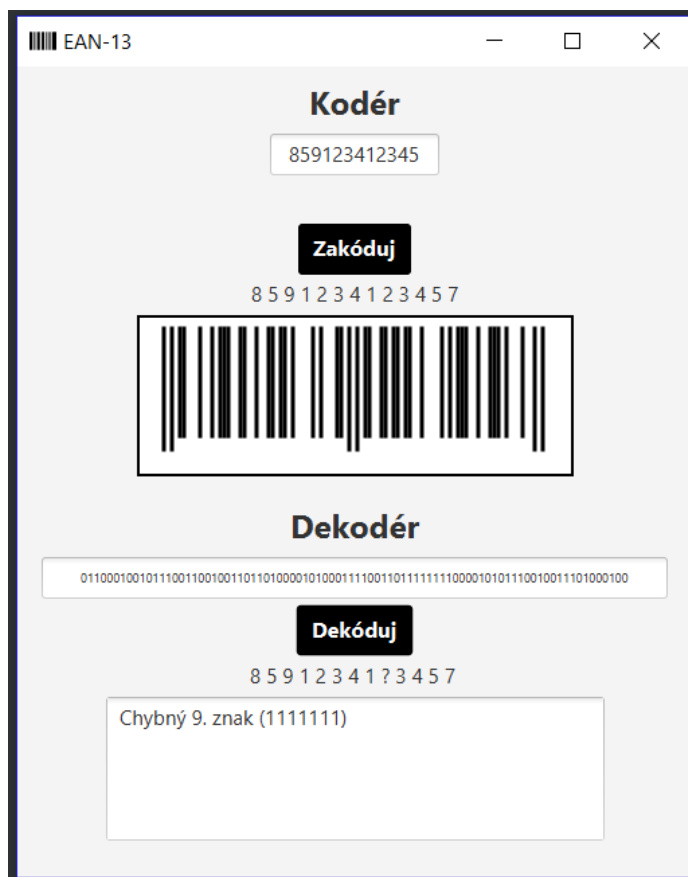
Aplikace je vyvíjena pomocí technologie JavaFX. Hlavní třída Main, pouze spouští aplikaci a vytvoří okno s uživatelským rozhraním načtené z fxml souboru. O veškerou funkcionalitu gui se stará třída MainController. Jednotlivé kódovací sady jsou implementované pomocí hashovacích tabulek. Tyto tabulky jsou v abstraktní třídě Aean13, kterou dědí třídy Coder a Decoder. Třída Coder se stará o funkci kodéru a třída Decoder o funkci dekodéru.

Uživatelské rozhraní

Celá aplikace je tvořena jedním jediným oknem. Kodér i dekodér mají vlastní vstupní textové pole, ve kterém je zadán vstup pro kódování či dekódování. Není-li zadán správný počet vstupních znaků, tlačítka „Zakóduj“ či „Dekóduj“ jsou nepoužitelná.

Při použití tlačítka „Zakóduj“ se hned pod dané tlačítko vypíše celý kód i s kontrolní číslicí a zároveň se zobrazí grafická reprezentace zakódovaného čárového kódu, jehož binární reprezentace se nastaví do vstupního textového pole pro dekódování.

Při použití tlačítka „Dekóduj“ se hned pod dané tlačítko vypíše dekódovaný čárový kód. Pokud se během dekódování vyskytly nějaké chyby, zobrazí se v textové oblasti pro výpis chyb. Na pozici znaku, kde byla detekována chyba, je zobrazen otazník.



Závěr

Student Jiří Bešta se zaměřil především na tvorbu uživatelského rozhraní a student Jan Matušík se postaral o samotnou implementaci kodéru a dekodéru, avšak jejich práce se různě doplňovala.

Výsledkem je program s jednoduchým uživatelským rozhraním. Mezi možná vylepšení aplikace by se dalo zařadit např. export čárového kódu do některého z obrázkových formátů, rozpoznávání více druhů čárových kódů, ...