

Dokumentace

Samoreplikující script

“selfwriter”

Matěj Holubec, 20. 3. 2024

1. Stručné zadání (anotace)

Tento program je samoreplikující se script, který vytváří novou verzi sebe sama. Při spuštění generuje soubor s identickou funkcionalitou jako má původní skript. Program demonstruje koncepty sebereferece a serializace kódu.

2. Přesné zadání

Program vytváří kopii svého vlastního zdrojového kódu v novém souboru s příponou "new.py" přidanou k původnímu názvu souboru. Skript musí:

- Uložit svůj vlastní kód jako data uvnitř sebe
- Vygenerovat nový soubor, který je funkčně identický s originálem
- Zobrazit vygenerovaný kód v konzoli

3. Zvolený algoritmus

Algoritmus používá techniku sebereferece, kdy je kód programu uložen jako seznam řetězců. Klíčovým poznatkem je, že skript potřebuje reprezentovat sám sebe uvnitř tohoto seznamu.

Jednotlivé komponenty lze popsat následovně:

- Uložení všech řádků kódu jako řetězcové literály v seznamu
- Vložení reprezentace tohoto seznamu do sebe sama (rekurzivní definice)
- Zapsání všech řetězců do nového souboru
- Výpis všech řetězců v konzoli

4. Diskuse výběru algoritmu

Vytvoření samoreplikujícího se programu představuje zajímavou výzvu. Zvažovány byly tyto přístupy:

Čtení vlastního zdrojového souboru: Program by mohl přečíst svůj vlastní zdrojový soubor a zkopírovat ho. Toto řešení bylo zamítnuto, protože by nedemonstrovalo skutečnou seberefereci a autor nabýval silného dojmu, že by nebylo akceptováno pro účely zápočtového programu. Zároveň pro něj netvořilo výzvu

Přístup s řetězcovými literály (zvolený): Ukládání kódu jako řetězcových literálů činí sebeferenci explicitní a srozumitelnou, což je důvod, proč byl tento přístup vybrán.

5. Program

Hlavní datové struktury

- `print_statements`: Seznam řetězců, který obsahuje každý řádek kódu, který má být zapsán do nového souboru
- `SUFFIX`: Konstantní řetězec definující příponu, která se přidá k názvu generovaného souboru

Struktura programu

Program je organizován do funkce `main()`, která:

- Vloží samoreflexivní řádek do seznamu `print_statements`
- Otevře nový soubor (`__file__ + SUFFIX`)
- Zapiše všechny řádky do tohoto souboru
- Vypíše každý řádek na konzoli

6. Reprezentace vstupních dat

Tento program nevyžaduje žádná externí vstupní data. Je zcela samostatný.

7. Reprezentace výstupních dat

Program produkuje dva typy výstupu:

- Výstup do souboru: Vytváří nový Python soubor pojmenovaný podle původního souboru s přidaným "new.py" (např. pokud je původní soubor "main.py", vytvoří se "main.pynew.py").
- Výstup v konzoli: Zobrazuje každý řádek kódu, který byl zapsán do souboru.

Interpretace výstupu není potřeba, protože výstupem je jednoduše zdrojový kód sám.

8. Pokyny ke spuštění

1. Otevřete terminál/příkazový řádek
2. Navigujte do adresáře obsahujícího soubor
3. Spustěte příkaz: `python main.py`
4. Po spuštění najdete nový soubor s příponou "new.py" ve stejném adresáři. Proces můžete zopakovat s novým názvem v kroku 3

9. Poznámky pro programátory

Skript používá několik pozoruhodných technik a pomůcek:

- Escapování řetězců: Používá "surové" řetězce (r"...") pro vyhnutí se složitým escape sekvencím
- Unicode escapování: Používá \042 k reprezentaci uvozovek, aby se vyhnul problémům s uvozovkami při vkládání definice seznamu
- Reference na cestu k souboru: Používá `__file__` k odkazu na vlastní umístění

10. Sada testovacích příkladů

Testovací případ 1: Základní spuštění

Akce: Spuštění skriptu

Očekávaný výsledek: Měl by být vytvořen nový soubor s identickým kódem

Ověření: Spusťte nově vytvořený soubor a potvrďte, že vytvoří další identický soubor

Testovací případ 2: Více generací

Akce: Spusťte skript, poté spusťte vygenerovaný skript, poté spusťte ten vygenerovaný skript

Očekávaný výsledek: Každá generace by měla produkovat identické kopie

Ověření: Porovnejte obsah souborů pomocí nástroje diff, abyste zajistili, že jsou identické

11. Co nebylo doděláno

Současná implementace má několik omezení, která by mohla být řešena či pozměněna v budoucích verzích:

- Název souboru se s každou generací prodlužuje (přidává se "new.py" při každém spuštění)
- Nejsou implementovány argumenty příkazové řádky pro přizpůsobení názvu výstupního souboru

12. Závěrečná promluva

Tento zdánlivě jednoduchý skript poskytuje zdatnou ukázkou sebereferece v programování. Přestože je malý svým rozsahem, efektivně znázorňuje, jak kód může reprezentovat a reprodukovat sám sebe.