Dokumentace úlohy CHA: C Header Analysis v PHP 5 do IPP 2013/2014
Jméno a příjmení: David Kovařík
Login: xkovar66

Analýza zdrojových souborů

Úkolem programu je zpracování hlavičkových souborů jazyka C. Program vychází z předpokladu, že zkoumané soubory jsou zapsány podle normy ISO C99 v kódování utf-8. Implementace programu kombinuje několik programovacích paradigmat (která PHP umožňuje). Hlavní přístupový bod (spoštěná část) programu tedy spíše připomíná skript, ovšem pro svou činnost využívá několik modulů, které obsahují jak definici funkcí, tak tříd.

Zpracování souborů je prováděno v několika navazujících fázích.

Zpracování argumentů

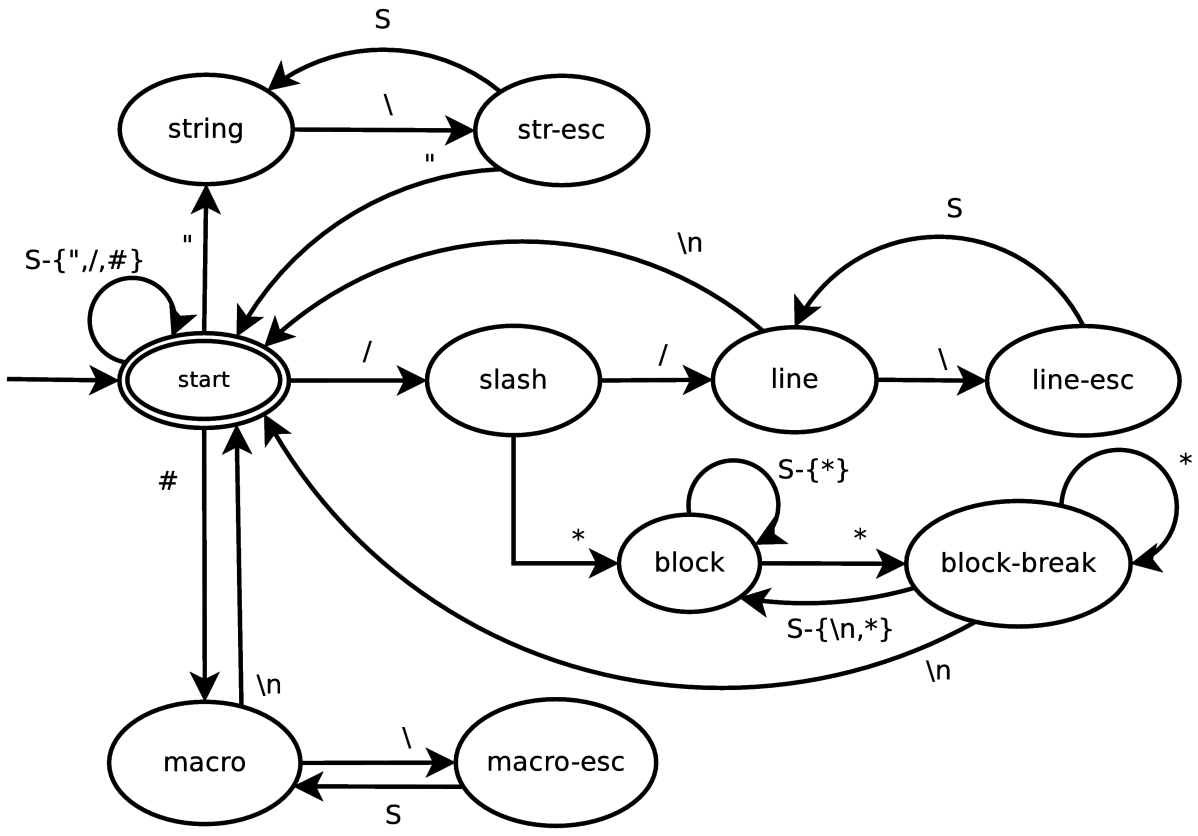
Během této fáze jsou inicializovány kontrolní struktury nesoucí informace o konfiguraci programu. Během samotného zpracování argumentů se vychází z předpokladu, že každý argument přímo vstupuje do konfigurace programu i v případě, kdy není explicitně zadán při spouštění programu. V takovém případě nabývá implicitní hodnoty. Příkladem může být argument –output=“... “. Pokud není explicitně zadán, bude výtupní soubor nastaven na standardní výstup. V průběhu programu pak není potřebat provádět další (zbytečné) kontroly nastavení této hodnoty.

Během této fáze také dochází k vygenerování seznamu pracovních souborů. Aby byla zachována jednotnost a jednoduchost, je i jeden explicitně zadaný soubor chápán jako kolekce (obsahující právě jeden prvek). Tento přístup opět umoňuje eliminaci nadbytečných kontrol.

Filtrování nepodstatných informací

V hlavičkových souborech se mohou vyskytovat také komentáře, makra či textové řetězce. Ty musí být ještě před samotnou analýzou odstraněny, jelikož by mohly způsobit neplatná data na výstupu programu. Nalézt vhodné regulární výrazy, které je odstraní by bylo značně složité, především kvůli možné kolizi v zanoření klíčových sekvencí znaků. Proto je filtrování prováděno pomocí jednoduchého konečného automatu, kde je prováděná redukce více transparentní.

Automat v počátečním stavu opisuje přečtený znak na výstup. Ostatní stavy provádí redukci, tedy vstupní znaky neopisují. Samotné analýze pak podléhá právě tímto způsobem redukovaný soubor. Schéma použitého automatu je znázorněno na následujícím obrázku.



Poznámka: symbol 'S' značí vstupní abecedu

Samotná analýza hlavičkových souborů

Hledání deklarací funkcí ve souborech je již prováděno pomocí regulárních výrazů. Nejdříve jsou v souborech nalezeny deklarace jako celek. Tyto shody jsou vyjmuty a uloženy do samostatných řetězců. Dále se s obsahem celého souboru již nemusí pracovat, hlubší analýze podléhají pouze vytvořené textové řetězce.

Opět pomocí regulárních výrazů je z každého řetězce přečten návratový typ a jméno dané funkce. Formální parametry jsou opět zpracovány samostatně. Pokud by počet parametrů přesahoval maximální povolený počet zadaný při spouštění programu, signalizuje analýza parametrů, aby aktuálně zpracovávaná funkce nebyla přidána do celkové hierarchie.

Formátování výstupu

V poslední fázi jsou zpracovaná data předána na výstup formou XML souboru. Pro reprezentaci každého typu značek je vytvořena samostatná třída, která zapouzdřuje potřebná data a přebírá zodpovědnost za převádění dat na textový výstup. Instance těchto tříd jsou v průběhu analýzy souboru dynamicky vytvářeny a případně zanořovány do očekávané hierarchie. K projití celé hierarchie značek tedy stačí pouze reference na hlavní element.

Formátování výstupu řeší samostatné třídy, kterým je předán právě jeden hlavní prvek. Tyto třídy pak projdou celou hierarchií a vytvoří formátovaný výstup. Jednolivé instance formátovacích tříd je pak možné snadno přepínat na základě konfiguračních struktur.