

Skaner HTML-a

1. WSTĘP.

Celem niniejszego projektu jest napisanie prostego “tłumacza” HTML-a do plików tekstowych. Waszym celem będzie użyć generatora skanerów *lex* lub *flex* do napisania programu, który “tłumaczy” HTML na tekst, nie będziemy przy tym próbować ogarnąć wszystkich subtelności języka HTML w taki sposób, w jaki powinno byłoby to być zrobione w komercyjnym programie.

Dokumentację do *lex*’a/*flex*’a znajdziecie tutaj:

<http://www.kompilatory.agh.edu.pl/pages/tk-laboratorium/flex.html>

1.1. Opis działania programu. Wasz program powinien pobierać dane z *stdin*, wyrzucać z nich wszystkie tagi HTML (w tym komentarze, patrz opis poniżej), rozpoznawać i obrabiać pewien zestaw “znaków specjalnych” i zapisywać wynik do *stdout*.

1.2. Tagi HTML. HTML standardowo definiuje szeroką gamę tagów. Jako że naszym głównym celem jest nauczenie się posługiwania *flex*’em, uprościmy sobie zadanie przyjmując następującą definicję tagu:

Tagiem nazywamy dowolny ciąg znaków postaci $< S >$, gdzie S jest ciągiem drukowalnych znaków, który nie zaczyna się od “białych spacji” i nie zawiera znaku $>$.

Znaki drukowalne rozumiemy w sensie, w jakim specyfikuje je funkcja *isprint()* w C, a “białe spacje” w sensie, w jakim specyfikuje je funkcja *isspace()*. Odpowiadają one klasom znaków *flex*’a definiowanym przez *[::print:]* oraz *[::blank:]*, odpowiednio.

Przykładowo, każdy z następujących ciągów znaków jest tagiem:

- $< b >$
- $< br >$
- $< ahref = "http://www.math.us.edu.pl/pgladki/" >$
- $< /b >$
- $< !- \text{this is an HTML comment, and has the structure of a tag as described above} - >$

1.3. Znaki specjalne. HTML definiuje pewne “znaki specjalne”, które Wasz program powinien umieć rozpoznawać i zamieniać według następującego schematu:

- *&*; zamieniać na *&*
- *<*; zamieniać na *<*
- *>*; zamieniać na *>*
- *"*; zamieniać na *"*

2. WYWOŁYWANIE PROGRAMU.

Plik wykonywalny powinien się nazywać *myhtml2txt* i powinien czytać z pliku *stdin* i zapisywać do pliku *stdout*. Innymi słowy, “tłumaczenie” pliku *foo.html* do *foo.txt* powinno być wywołane poleceniem:

myhtml2txt < foo.html > foo.txt