Obligatorisk oppgave 1
-
Kort om oppgaven og verktøyene

Fredrik Sørensen
OMS-gruppen, IfI

Oversikt

- Litt om meg
- Praktisk
- Oppgaven
- Verktøyene
  - Ant
  - JFlex
  - CUP
- Utfordringer
- Torsdag

Litt om meg

- Fredrik Sørensen
  - Stipendiat I SWAT-prosjektet (OMS)
    - Veileder er Stein Krogdahl
    - Stipendiat siden januar 2006
    - Forsker på programmeringsspråk
  - Master UiO, 2005
  - Tok INF 5110 (INF310) våren 2002
  - Grupperlærer i INF 5110 i fjor også
  - Underviste INF 1000 i høst

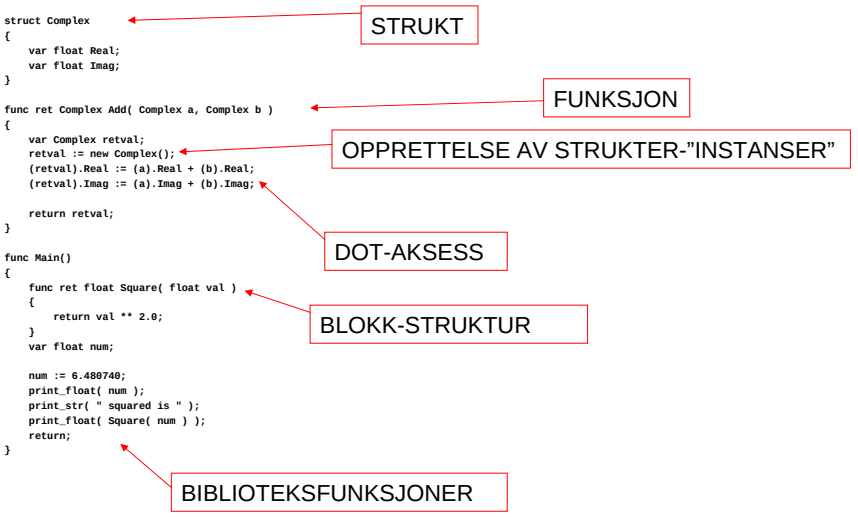
Praktisk

- Oblig 1 legges ut på kurssiden etter forelesningen (Vær obs på endringer).
- Grupper på inntil 3 personer
- Frist
  - Fredag 28 mars (1 uke etter langfredag/påsken)
- Jeg kan nås på e-post, telefon og på kontoret
  - Epost: fredrso@ifi.uio.no
  - TLF: 22 85 04 73
  - Kontor: 4311-NR (4. etasje)

Oppgaven

- Bruke skannings- og parseringsverktøy
  - JFlex
  - CUP
- Omarbeide en grammatikk fra én form til en annen
  - Utvidet BNF
  - BNF som passer verktøyet
- Kunne kontrollere presedens og assosiativitet på to måter:
  - Ved å lage en entydig grammatikk som gir riktig presedens/assosiativitet
  - Gi en flertydig grammatikk og styre presedens og assosiativitet ved passelige kommandoer i CUP.
- Designe noder til et passelig abstrakt syntaks-tre
- Bruke parserverktøyet slik at man får bygget opp trærne
- Skrive det abstrakte syntakstreet ut i et gitt format
- Rapporten teller!

D_b (D-flat)



Pass By Reference

```
void swap( ref int a, ref int b ) {
  int tmp = a;
  a = b;
  b = tmp;
}

void Main() {
  int x = 42;
  int y = 84;
  swap( ref x, ref y );
}
```

Terminaler

- **NAME**
  - skal starte med bokstav, og deretter være en sekvens av siffer, bokstaver og underscore. De kan ikke være på mer enn 16 tegn, og alle er signifikante. Store og små bokstaver regnes som forskjellige tegn. Merk altså at alle nøkkelord skrives med små bokstaver, og at de ikke kan brukes som vanlige navn.
- **INT_LITERAL**
  - skal inneholde ett eller flere siffer.
- **FLOAT_LITERAL**
  - skal inneholde ett eller flere siffer, fulgt av et punktum, fulgt av ett eller flere siffer.
- **STRING_LITERAL**
  - skal bestå av en tekststreng innesluttet i anførselstegn ("). Strengen kan ikke inneholde linjeskift. Den semantiske "verdien" av en STRING_LITERAL er kun det som er inni anførselstegn; selve anførselstegnene skal ikke inkluderes.

Metasymboler

- {…}
  - gjentakelse null eller flere ganger
- […]
  - betyr at det kan være med eller ikke

Grammatikken

```
PROGRAM      -> DECL { DECL }

DECL         -> VAR_DECL | FUNC_DECL | STRUCT_DECL

VAR_DECL     -> "var" TYPE NAME ";"

FUNC_DECL    -> "func" [ "ret" TYPE ] NAME "(”
                        [PARAM_DECL {
                            "," PARAM_DECL }
                        ] ")”
                        "{" { DECL } { STMT } "}"

STRUCT_DECL  -> "struct" NAME "{" { VAR_DECL } "}"

PARAM_DECL   -> [ "ref" ] TYPE NAME
```

Gramatikken

```
EXP      -> EXP LOG_OP EXP
         | "!" EXP
         | EXP REL_OP EXP
         | EXP ARIT_OP EXP
         | "(" EXP ")"
         | LITERAL
         | CALL
         | "new" NAME "(" ")"
         | VAR

VAR      -> NAME | EXP "." NAME

LOG_OP   -> "&&" | "||"

REL_OP   -> "<" | "<=" | ">" | ">=" | "=" | "!="

ARIT_OP  -> "+" | "-" | "*" | "/" | "**"

LITERAL  -> FLOAT_LITERAL | INT_LITERAL | STRING_LITERAL
         | "true" | "false" | "null"

STMT     -> ASSIGN_STMT
         | IF_STMT
         | WHILE_STMT
         | RETURN_STMT
         | CALL_STMT
```

Gramatikken

```
ASSIGN_STMT  -> VAR ":=" EXP ";"

IF_STMT      -> "if" EXP "then" { STMT }
                [ "else" { STMT } ] "endif"

WHILE_STMT   -> "while" EXP "do" "{" { STMT } "}"

RETURN_STMT  -> "return" [ EXP ] ";"

CALL_STMT    -> CALL ";"

CALL         -> NAME "(" [ ACTUAL_PARAM { ",”
                           ACTUAL_PARAM } ] ")"

ACTUAL_PARAM -> "ref" VAR | EXP

TYPE         -> "float" | "int" | "string" | "bool" | NAME
```

Verktøyene

- Ant
- JFlex
- CUP

Ant

- Taskdef

```
<taskdef classname="JFlex.anttask.JFlexTask" name="jflex"
  classpathref="path-cup" />
<taskdef classname="java_cup.anttask.CUPTask" name="cup"
  classpathref="path-cup"/>
```

- Generate

```
<target name="generate" depends="mkdir">
    <jflex file="./grammars/oblig1.lex" destdir="src-gen"/>
     <cup srcfile="./grammars/oblig1.cup" destdir="src-gen“
          interface="true" />
</target>
```

Ant

- Compile

```
<target name="compile" depends="generate">
     <javac srcdir="${dir.src}" destdir="${dir.classes}“
          debug="true" includes="**/*.java“
          classpathref="path-cup" sourcepath="${dir.src-gen}"/>
     <javac srcdir="${dir.src-gen}" destdir="${dir.classes}"
          debug="true" includes="**/*.java"
          classpathref="path-cup“ sourcepath="${dir.src}" />
 </target>
```

- Run

```
<target name="run" depends="init">
    <java classname="compiler.Compiler“
        classpathref="path-run">
         <arg value="code-examples/Canonical.d"/>
         <arg value="code-examples/Canonical.ast"/>
    </java>
 </target>
```

JFlex

```
package oblig1parser;
import java_cup.runtime.*;

%%

%class Lexer
%unicode
%cup
%line
%column
%public

%{
%}

WhiteSpace= [ \t\f]

%%

{WhiteSpace}{ }
"struct"{ return new Symbol(sym.STRUCT); }
"{"{ return new Symbol(sym.LBRACK); }
"}"{ return new Symbol(sym.RBRACK); }

.     { throw new Error("Illegal character '" + yytext() + "' at line " +
   yyline + ", column " + yycolumn + "."); }
```

Brukerkode

Opsjoner og deklarasjoner

Leksikalske regler

CUP

```
package oblig1parser;
import java_cup.runtime.*;
import syntaxtree.*;
import java.util.*;

parser code {:
:};


/* Terminals */
terminal            STRUCT;
terminal LBRACK, RBRACK;

/* Non terminals */
non terminalProgramprogram, struct_decl;


/* The grammar */
program     ::= struct_decl:sd {: RESULT=sd; :}
              ;
struct_decl ::= STRUCT LBRACK RBRACK {: RESULT=new Program(); :}
              ;
```

Brukerkode

Opsjoner og deklarasjoner

Actions

Gramatikken

Utfordringer

- Lage de to grammatikkene og løse skift/reduser-konflikter
- Bruke parserverktøyet slik at man får bygget opp trærne

Torsdag

- Flere detaljer om CUP og JFlex
  - Produksjoner
  - Stacken
  - Assosiativitet og presedens
- Spørsmål
  - Send meg noe så snart det er noe dere lurer på
- Legger ut oblig-filer i dag!

SPØRSMÅL ???