

Compilers - Encyclopedia Academia

 tuyaux.winak.be/index.php/Compilers

Compilers

Richting Informatica

Jaar 2BINF

Bespreking

Zoals de naam doet vermoeden wordt in dit vak uitgelegd wat een compiler doet, gaande van het inlezen van de broncode tot het genereren van machinecode. Er is een project voor dit van waarin je een subset van C moet implementeren met behulp van een compiler compiler. Op het moment van schrijven was dit ANTLR v4. Bij de verdediging wordt dan ook je kennis van de theorie getoetst.

Vanaf 2018-2019 wordt dit vak gegeven door Guillermo. Als je ervoor gaat, is een 20/20 halen bij hem zeker mogelijk. Je kan ook een presentatie over een onderwerp van compilers doen in een middelbare school voor 2 bonuspunten. Op moment van schrijven was zijn cursus wel nog niet af en de slides heel beknopt, dus let best op tijdens de lessen.

Puntenverdeling

In 2012 - 2013 stond het schriftelijk examen op 30% van het totaal; het project viel tijdens het tweede semester en stond op de rest van de punten.

Examenvragen

Academiejaar 2020- 2021 - 1e zittijd

Je kan de examenvragen raadplegen in dit document: [Bestand:INF-Compilers-2021.1.pdf](#)

Academiejaar 2019- 2020 - 1e zittijd

Je kan de examenvragen raadplegen in dit document: [Bestand:INF-Compilers-2020.1.pdf](#)

Academiejaar 2018 - 2019 - 2e zittijd

Je kan de examenvragen raadplegen in dit document: [Bestand:BINF-Compilers-2019.2.pdf](#)

Academiejaar 2018 - 2019 - 1e zittijd

Academiejaar 2017 - 2018 - 1e zittijd

Theorie

1. Compileer het volgende assignment naar P-code:

```
k := pt↑.b*j
```

waarbij de volgende decaraties gelden:

```
type t = record
  a: integer
  b: integer
end;
var k, j: interger;
    pt: ↑t;
```

en $p(k) = 5$. Als je extra veronderstellingen maakt over p, vermeld die dan expliciet.

2. Laat G de volgende CF grammatica zijn:

```
S → E          T → FR
E → TK        R → λ | *T
K → λ | +E    F → (E) | id
```

- Geef First(KR) en Follow(T)
- Is deze grammatica geschikt voor recursive descent parsing? Motiveer!

3. Laat G de volgende CF grammatica zijn:

```
S → Aa | BAc | dc | bda
A → d
B → b
```

- Geef de LR(0) Automaat voor G
- Is de grammatica SLR(1)? Waarom?
- Is de grammatica LALR(1)? Waarom?

4. Abstracte Interpretatie

- Verklaar kort de regel $[[x \leftarrow M[e]]]^\# A = A \setminus \text{Occ}(x)$ voor het berekenen van sets van beschikbare assignments
- Op pag. 26 van reeks 9 (beschikbare assignments) wordt de hele set van mogelijk beschikbare assignments gebruikt asl 'bottom' element van \subseteq . Die hele set bestaat in dit geval uit slechts één element. Wat is, in het algemeen, de set van mogelijk beschikbare assignments? Welke vorm moeten die hebben en waarom?
- Tabel van pag. 28: Wat is de F van pag. 25 in dit geval?

5. Pag. 11 van reeks 10, procedure TreeCodeGenerate (TreeCG). Leg precies uit wat de rol van de tweede parameter is (type, betekenis).

Academiejaar 2016 - 2017 - 1e zittijd

Theorie

1. Codegeneratie voor shortcut - evaluatie van boolean expressies.
 1. Leg kort de rol uit van de 4 attributen.
 2. Geef de regel voor **xor** naar analogie met die voor **or** en **and**.
 3. Geef de code gegenereerd voor *if not (a xor b) S₁ else S₂*, waarbij S₁ en S₂ niet verder zijn uitgewerkt.
2. Beschouw de CFG:
S --> Aa | bAc | dc | bda
A --> d
 1. Geef de canonieke LR(0) automaat.
 2. Is deze grammatica geschikt voor SLR(1) parsing? Waarom (niet)?
 3. Is deze grammatica geschikt voor LALR(1) parsing? Waarom (niet)?
3. Analyse voor liveness (slides in cursus te vinden)
 1. (zie slide) Hoe ziet de tabel er uit als je de states doorloopt in volgorde 5,4,3,7,6,2,1,0 i.p.v. 7,6,2,5,4,3,1,0
 2. (zie slide) Hoe ziet ze er uit voor dit programma als x verondersteld wordt live te zijn op het eind, en volgorde 7,6,2,5,4,3,1,0?
4. Analyse voor constant folding (slide deel 9 slide 42) Leg uit wat de items in de rij met nummer 5 betekenen.
5. Scheduling algoritme (slide: deel 10 slide 20): Leg uit waarom de tweede en derde gevallen van "select" redelijke keuzen zijn.

Academiejaar 2012 - 2013 - 1e zittijd

Theorie

1. Compile the assignment
 $k := (15 - a) + b$
where k , a and b are integer variables and $p(k)=5$, $p(a)=9$, $p(b)=10$.
2. Explain briefly why the descriptor of a dynamic array contains a "fictitious start address" and not just the real start address of the block of array components.

3. Consider the program

```
program main;
var a, b: integer; var t1: array[0..7, 1..25] of integer;
proc pr(value tab: array[r1..h1,r2..h2] of integer);
begin
  ...
  tab[a,b+7] := 18;
end;
...
pr(t1);
...
end;
```

Give the compilation for the assignment

```
tab[a,b+7] := 18;
```

where $p(tab)=(105,1)$, $p(a)=(9,0)$, $p(b)=(10,0)$.

4. Compute the LR(0) characteristic automaton for the following grammar. Is the grammar suitable for LR(0) parsing?

```
P -> begin Ds ; Sts end
Ds -> Ds ; d | d
Sts -> Sts ; s | s
```

5. In Chaitin's algorithm for register allocation by graph coloring, why is it reasonable to compute BestCost by diving Cost(n) by NeighbourCount(n).

Academiejaar 2006 - 2007 - 2de zittijd

Theorie

Het theorie-examen is open boek.

1. Leg in eigen woorden uit hoe de statische link wordt ingevuld bij creatie van het stackframe. M.a.w. welke informatie hebben we nodig? Waar vinden we dat en wat gebeurt ermee?
2. Waarom kunnen er in een LR-automaat geen shift-shift conflicten voorkomen?
3. Wat betekent het dat een symbolisch register live is? Belang?
4. Attributengrammatica geven van een gegeven grammatica.

| E -> E+T | T T -> T * F | F F -> (E) | id