

TÖL304G Miðmisserispróf

TÖL304G Midterm Exam

1. Write on these pages and **do not reorder** the pages.
2. If the answer does not fit in the space intended then you may write on the empty pages at the end, but then you must say so in the space intended for the answer. For example “framhald á síðu 13”. or “continued on page 13”.
3. Refrain from folding or tearing these pages, they will have to go through a scanner. Write clearly with **dark ink or dark pencil** and do not write on the margins.
4. The backs of the pages **will not be scanned** and can be used for scratch. Any answers written on the backs **will be ignored**.

Nafn/Name:

Háskólatölvupóstfang/University email:

1. (5%) Sýnið BNF, EBNF, samhengisfrjálsa mállýsingu (CFG) eða málrit fyrir mál strengja yfir stafrófið $\{a, (,)\}$ þar sem svigar eru í jafnvægi.

Show BNF, EBNF, a context free grammar (CFG), or syntax diagrams for the language of strings over the alphabet $\{a, (,)\}$, where parentheses are balanced.

Dæmi um strengi í málinu. Examples of strings in the language.

ϵ (tómi strengurinn, the empty string)

a

aaa

$()$

$((a))$

$a(aa(aa)a)aa$

Dæmi um strengi ekki í málinu. Examples of strings not in the language.

$()$

$()$

$()$

$(a$

$)$

Svar:

2. (10%)

- a. Íhugið eftirfarandi BNF skilgreiningu. Consider the following BNF definition.

$$\langle E \rangle ::= p \langle E \rangle q \langle E \rangle \mid r$$

Hverjir eftirfarandi strengja eru í málinu sem skilgreint er? Tvö röng svör gefa núll stig. Vísbending: p og q eru í jafnvægi.

Which of the following strings are in the language defined? Two wrong answers give zero points. Hint: p and q are in balance.

- | | |
|-----------|--------------|
| 1. r | 6. $qrpr$ |
| 2. p | 7. pqr |
| 3. q | 8. $pprqrqr$ |
| 4. prq | 9. $prqprqr$ |
| 5. $prqr$ | 10. $pqprqr$ |

Svar:

- b. Hverjar af eftirfarandi reglulegu segðum skilgreina mál þar sem a og b eru í jafnvægi, þ.e. strengi þannig að ef öllum ' a ' væri breytt í ' $($ ' og öllum ' b ' breytt í ' $)$ ' þá væru svigar í jafnvægi?

Tvö röng svör gefa núll stig.

Which if the following regular expressions define a language where a and b are balanced, i.e. strings such that if all ' a ' were to be changed to ' $($ ' and all ' b ' changed to ' $)$ ' then parentheses would be balanced? Two wrong answers give zero points.

- $a^i b^i$
- $(ab)^i$
- $a(bbaa)^i b$
- $a(ba)^i b$
- $(a(a(ab)^i b)^i b)^i$
- $aa(aabb)^i bbab$
- $(ab|a|b)^i$
- $(ab|a(ab)^i b)^i$

Svar:

3. (15%)

Hverjar eftirfarandi fullyrðinga um lokanir eru sannar? Tvö röng svör gefa núll punkta.

Which of the following statements about closures are true? Two wrong answers give zero points.

- a) Lokanir innihalda fallsbendi. Closures contain a function pointer.
- b) Lokanir má nota til að útfæra strauma í Scheme. Closures can be used to implement streams in Scheme.
- c) Lokanir innihalda tengihlekk (aðgangshlekk). Closures contain an access link (static link).
- d) Lokanir innihalda stýrihlekk. Closures contain a control link (dynamic link).
- e) Lokanir innihalda straum. Closures contain a stream.
- f) Lokanir eru notaðar til að skila staðværu falli sem gildi í bálkmótuðum forritunarmálum. Closures are used to return a local function as a value in block structured programming languages.
- g) Lokanir eru til í C. Closures exist in C.
- h) Lokanir eru til í Scheme. Closures exist in Scheme.
- i) Lokanir eru til í CAML. Closures exist in CAML.
- j) Lokanir eru aðeins mögulegar ef vakningarfærslur eru í kös. Closures are only possible if activation records are in the heap.

Svar:

4. (15%)

Skrifið eftirfarandi Scheme fall. Notið aðeins einfaldar Scheme aðgerðir eins og í dæminu á eftir. Write the following Scheme function. Use only simple operations as in the next question.

```
;; Notkun: (foldl f u x)
;; Fyrir:  f er tvíundarfall,
;;         u er eitthvert gildi,
;;         x=(x1 ... xN) er listi einhverra gilda
;; Gildi:  (f (f ... (f u x1) ...) xN)
;; Ath.:   (foldl f u '()) skal skila u
```

```
;; Usage:  (foldl f u x)
;; Pre:    f is a binary function,
;;         u is some value,
;;         x=(x1 ... xN) is a list of some values
;; Value:  (f (f ... (f u x1) ...) xN)
;; Note:   (foldl f u '()) should return u
```

Svar:

5. (15%) Skriðið **halaendurkvæmt** fall í Scheme, CAML eða Morpho sem (í Morpho) hefur eftirfarandi lýsingu. Ekki má nota lykkju og ekki má gefa breytu nýtt gildi. Notaðu hjálparföll sem þú þurfið þá að skrifa sjálf **ásamt lýsingu þeirra**. Aðeins má nota einfaldar innbyggðar aðgerðir í Scheme, CAML og Morpho, þ.e. aðgerðir sem hafa tímaflækju $O(1)$ svo sem cons, car, cdr og null?.

Write a **tail recursive** function in Scheme, CAML or Morpho, which (in Morpho) has the following description. No loops are allowed and it is not allowed to give a variable a new value. Helper functions can be used, which you must then write yourself, **along with their descriptions**. Only simple built-in operations in Scheme, CAML and Morpho may be used, i.e. operations that have time complexity $O(1)$ such as cons, car, cdr and null?.

```
;;; Notkun: x = iota(n);  
;;; Fyrir:  n er heiltala, n>=0.  
;;; Gildi:  x er listinn [1,2,...,n]  
  
;;; Usage:  x = iota(n);  
;;; Pre:    n is an integer, n>=0.  
;;; Value:  x is the list [1,2,...,n]
```

Svar:

6. (15%)

Skrifið map fall í Scheme, CAML eða Morpho. Lýsingin í Scheme er eftirfarandi. Notið aðeins einfaldar aðgerðir eins og í fyrra dæmi.

Write a map function in Scheme, CAML or Morpho. The description in Scheme is the following. Use only simple operations as in the previous question.

```
;; Notkun: (map f x)
;; Fyrir:  f er einundarfall,
;;         x=(x1 ... xN) er listi.
;; Gildi:  Listinn (y1 ... yN) þar sem yI er
;;         útkoman úr kallinu (f xI),
;;         fyrir I=1,...,N
```

Svar:

7. (15%)

Skrifið eftirfarandi `powerList` fall í Scheme, CAML eða Morpho. Lýsingin í Scheme er eftirfarandi. Þið megið reikna með því að til sé fall `append` sem skeytir saman tveimur listum ásamt `map` falli fyrir lista með venjulegri lýsingu. Í Morpho má nota lykkjur, en þá þurfa þær að hafa fastayrðingar lykkju.

Write the following `powerList` function in Scheme, CAML or Morpho. The description in Scheme is the following. You may assume the existence of a function `append` that concatenates two lists as well as a `map` function for lists with the usual description. In Morpho you may use loops, but then they must have loop invariants.

```
;; Notkun: (powerList n)
;; Fyrir:  n er heiltala, n>=0.
;; Gildi:  Listi allra undirlista listans
;;         (n n-1 ... 3 2 1).
;;         Þetta er listi sem inniheldur 2^n
;;         undirlista.

;; Usage:  (powerList n)
;; Pre:    n is an integer, n>=0.
;; Value:  The list of all sublists of the list
;;         (n n-1 ... 3 2 1).
;;         This is a list that contains 2^n
;;         sublists.
```

Svar:

8. (10%)

Vakningarfærsla falls í bálkmótuðu forritunarmáli eins og Scheme inniheldur sum eftirfarandi atriða. Hver? Eitt rangt svar gefur núll stig. The activation record (stack frame) of a function in a block-structured programming language such as Scheme contains some of the following. Which? One wrong answer gives zero points.

- a) Staðværar breytur fallsins. The local variables of the function.
- b) Víðværar breytur sem eru aðgengilegar í fallinu. The global variables that are accessible in the function.
- c) Viðföng fallsins. The arguments of the function.
- d) Staðværar breytur fallsins sem kallaði á fallið. The local variables of the function that called the function.
- e) Aðgangshlekk (tengihlekk). An access link (static link).
- f) Stýrihlekk. A control link (dynamic link).
- g) Vendivistfang. A return address.
- h) Benda á öll föll sem hægt er að kalla á úr fallinu. Pointers to all the functions that can be called from the function.
- i) Benda á allar lifandi vakningarfærslur. Pointers to all living activation records.
- j) Alla hluti sem til eru í kerfinu. All objects that exist in the system.
- k) Vakningarfærslur allra falla sem hægt er að kalla á. The activation records of all functions that can be called.
- l) Skráakerfi tölvunnar. The file system of the computer.
- m) Nöfn allra falla sem hægt er að kalla á. The names of all functions that can be called.

Svar:

(Aukablaðsíða / Extra page)

(Aukablaðsíða / Extra page)

(Aukablaðsíða / Extra page)

(Aukablaðsíða / Extra page)