

TÖL304G Final Exam

TÖL304G Lokapróf

Nafn/Name:

Háskólatölvupóstfang/University Email:

1. No help materials are allowed.
Engin hjálpargögn eru leyfileg.
2. Write your answers on these pages, not in an exam book.
Skrifið svörin á þessar síður, ekki í prófbók.
3. If the answer does not fit on the allotted space you may write on the empty pages at the end, but in that case you should write an indication to that effect on the space allotted for the answer, for example “continued on page 22”.
Ef svarið kemst ekki fyrir á tilteknu svæði má skrifa á auðar síður aftast, en þá skalt þú láta vita af því með því að skrifa tilvísun í tiltekið svæði, til dæmis “framhald á blaðsíðu 22”.
4. Refrain from mutilating or tearing these pages, they have to go through a scanner. Write clearly with **dark letters** and do not write in the margins.
Forðist að skemma eða rífa þessar síður, þær þurfa að fara gegnum skanna. Skrifið skýrt með **dökku lettri** og ekki skrifa í spássíur.
5. The backs of the pages **will not be scanned** and can be used for scratch. Any answers written on the backs **will be ignored**.

Baksíður **verða ekki skannaðar** og má nota fyrir krass. **Ekki verður tekið tillit til** svara sem skrifuð eru á baksíður.

6. The exam is divided into **parts**. Answer **10** questions in total and at least **the specified required number** from each of the parts.
Prófið skiptist í **hluta**. Svárið **10** spurningum í heild og að minnsta kosti **tilteknum lágmarksfjölda** í hverjum hluta.
7. If you answer more than **10** questions then your grade will be computed as the **average of all questions answered** unless you clearly **cross out** answers you do not want to count. You must cross out all the answer, not just part of it. Ef þú svarar fleiri en **10** spurningum þá verður einkunn þín reiknuð sem **meðaltal allra svara** nema þú **krossir skýrt út** svör sem þú vilt ekki að gildi. Þú verður að krossa út allt svárið, ekki aðeins hluta þess.
8. Remember that all functions need a **description** with Usage/Pre/Post or Usage/Pre/Value.
Munið að öll föll þurfa **notkunarlýsingu** með Notkun/Fyrir/Eftir eða Notkun/Fyrir/Gildi.
9. Remember to use proper **indentation** in all program code.
Munið að nota viðeigandi **innfellingu** í öllum forritstexta.

Part I – Block-Structure, etc.**Hluti I – Bálmótun o.fl.**

Answer at least two questions in this part – Remember to answer at least 10 questions in total

Svarið að minnsta kosti tveimur spurningum í þessum hluta – Munið að svara a.m.k. 10 spurningum í heild

1.

Hverjar eftirfarandi fullyrðinga um aðgangshlekkir (tengihlekkir), stýrihlekkir og lokanir eru sannar? Tvö röng svör gefa núll punkta. Which of the following statements about access links (static links), control links (dynamic links) and closures are true? Two wrong answers give zero points.

- a) Aðgangshlekkir eru notaðir í bæði bálmótuðum og öðrum forritunarmálum. Access links are used in both block structured and other programming languages.
- b) Stýrihlekkir eru notaðir bæði í bálmótuðum og öðrum forritunarmálum. Control links are used in both block structured and other programming languages.
- c) Lokanir innihalda aðgangshlekk. Closures contain an access link.
- d) Lokanir innihalda stýrihlekk og aðgangshlekk. Closures contain a control link and an access link.
- e) Lokanir innihalda vendivistfang og stýrihlekk. Closures contain a return address and a control link.
- f) Lokanir innihalda fallsbendi. Closures contain a function pointer.
- g) Aðgangshlekkir eru ekki til í Haskell. Access links do not exist in Haskell.
- h) Lokanir eru ekki til í Scheme. Closures do not exist in Scheme.
- i) Stýrihlekkir eru ekki til í CAML. Control links do not exist in CAML.
- j) Lokanir eru ekki til í Morpho. Closures do not exist in Morpho.
- k) Aðgangshlekkir eru ekki til í Java. Access links do not exist in Java.
- l) Lokanir eru aðeins mögulegar ef vakningarfærslur eru í kös. Closures are possible only if activation records are in the heap.

Svar/Answer:

a	b	C	d	e	f	g	h	i	j	k	l

2.

Vakningarfærsla falls í bálmótuðu forritunarmáli eins og Scheme inniheldur sum eftirfarandi atriða. Hver? Tvö röng svör gefa núll stig. Þið megið reikna með að ekki sé um halaendurkvæmni að ræða og að vakningarfærslan sé samhangandi minnissvæði.

The activation record (stack frame) of a function in a block-structured programming language such as Scheme contains some of the following. Which? Two wrong answers give zero points. You may assume that there is no tail-recursion and that the activation record is a contiguous memory area.

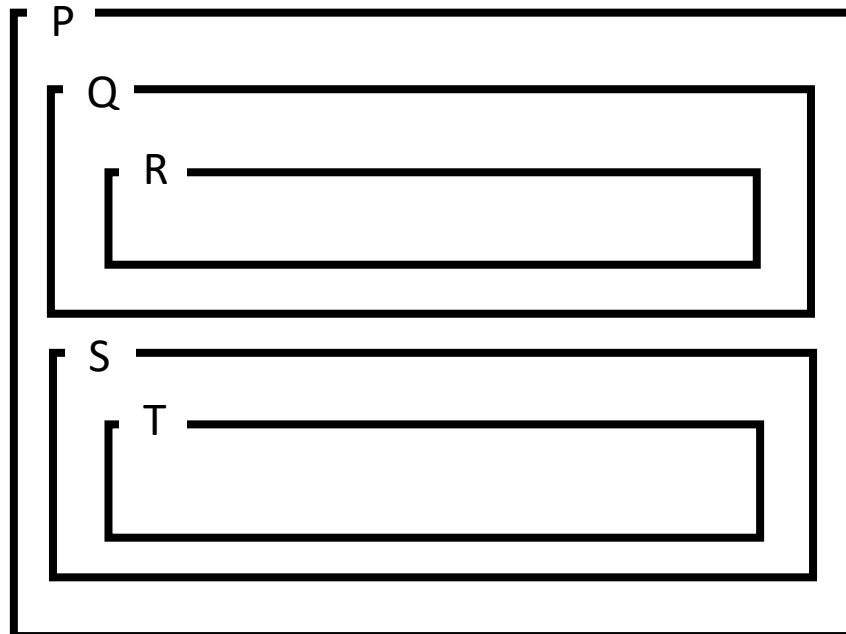
- a) Vendivistfang. A return address.
- b) Benda á öll föll sem hægt er að kalla á úr fallinu. Pointers to all the functions that can be called from the function.
- c) Skráakerfi tölvunnar. The file system of the computer.
- d) Benda á allar lifandi vakningarfærslur. Pointers to all living activation records.
- e) Stýrihlekk. A control link (dynamic link).
- f) Alla hluti sem til eru í kerfinu. All objects that exist in the system.
- g) Viðföng fallsins. The arguments of the function.
- h) Vakningarfærslur allra falla sem hægt er að kalla á. The activation records of all functions that can be called.
- i) Lokun sem vísar á fallið. A closure that refers to the function.
- j) Staðværar breytur fallsins. The local variables of the function.
- k) Bendi á vakningarfærslu fallsins sem kallaði á fallið. A pointer to the activation record of the function that called the function.
- l) Bendi á vakningarfærslu fallsins sem inniheldur fallið, textalega séð, ef eitthvert er. A pointer to an activation record of the enclosing function, if any.
- m) Aðgangshlekk (tengihlekk). An access link (static link).

Svar/Answer:

a	b	c	d	e	f	g	H	i	j	k	l	m

3.

Íhugið myndina sem sýnir földun falla P, Q, R, S og T.
Samsvarandi Scheme forritstexti er einnig sýndur í tveimur
jafngildum útgáfum hlið við hlið.



Consider the figure which shows nesting of functions P, Q, R, S and T. The corresponding Scheme code is also shown in two equivalent versions side by side.

```
(define (P ...)
  (define (Q ...)
    (define (R ...)
      ...[stofn R/body of R]
    )
    ...[stofn Q/body of Q]
  )
  (define (S ...)
    (define (T ...)
      ...[stofn T/body of T]
    )
    ...[stofn S/body of S]
  )
  ...[stofn P/body of P]
)
```

```
(define (P ...)
  (define (S ...)
    (define (T ...)
      ...[stofn T/body of T]
    )
    ...[stofn S/body of S]
  )
  (define (Q ...)
    (define (R ...)
      ...[stofn R/body of R]
    )
    ...[stofn Q/body of Q]
  )
  ...[stofn P/body of P]
)
```

Fyllið út eftirfarandi töflur með því að setja krossa við sannar fullyrðingar.
Eitt rangt svar gefur núll í einkunn fyrir dæmið.

Fill out the following tables by ticking the correct assertions. One wrong

answer gives zero points for the question.

Svar / Answer:

Kalla má á P úr:

P may be called from:

P	Q	R	S	T

Kalla má á Q úr:

Q may be called from:

P	Q	R	S	T

Kalla má á R úr:

S may be called from:

P	Q	R	S	T

Kalla má á S úr:

T may be called from:

P	Q	R	S	T

Kalla má á T úr:

T may be called from:

P	Q	R	S	T

Staðværar breytur í P má nota í:

Local variables in P may be used in:

P	Q	R	S	T

Staðværar breytur í Q má nota í:

Local variables in Q may be used in:

P	Q	R	S	T

Staðværar breytur í R má nota í:

Local variables in R may be used in:

P	Q	R	S	T

Staðværar breytur í S má nota í:

Local variables in S may be used in:

P	Q	R	S	T

4.

Eftirfarandi forritstexti er í einhverju ímynduðu forritunarmáli.

The following program text is in some imagined programming language.

```
void f(x,y)
{
    y = 3;
    print x,y;
    x = 2;
}
int i,a[10];
for( i=0 ; i!=10 ; i++ ) a[i]=i+1;
f(a[a[0]],a[0]);
print a[0], a[1], a[2], a[3];
```

Hvað skrifar þetta forrit (sex gildi í hvert skipti) ef viðföngin eru:

What does the program write (six values each time) if the parameters are:

a) Gildisviðföng / Passed by value:

b) Tilvísunarviðföng / Passed by reference:

c) Nafnviðföng / Passed by name:

Part II – List Processing, etc.**Hluti II – Listavinnsla o.fl.**

Answer at least two questions in this part – Remember to answer at least 10 questions in total

Svarið að minnsta kosti tveimur spurningum í þessum hluta – Munið að svara a.m.k. 10 spurningum í heild

5.

Skrifið fall í Scheme, CAML, Morpho eða Haskell sem tekur eitt viðfang sem er listi lista af fleytitölum milli 0 og 1 og skilar tölu sem er stærsta lággildi innri listanna, þ.e. stærst af þeim tölum sem fást þegar fundin er minnsta tala í hverjum innri lista. Þið skuluð reikna með því að hágildi í tóma menginu sé 0 og lággildi í tóma menginu sé 1. Munið fallslýsingar, eins og alltaf. Fallið þarf að skila viðeigandi gildi bæði fyrir tóman lista og fyrir lista sem einungis inniheldur tóma lista.

Write a function in Scheme, CAML, Morpho or Haskell that takes one argument that is a list of lists of floating point numbers between 0 and 1 and returns a number that is the largest minimum of the inner lists, i.e. the largest of the numbers generated by finding the smallest number in each inner list. You should assume that the maximum of the empty set is 0 and the minimum of the empty set is 1. Remember function descriptions, as always. The function must return an appropriate value both for the empty list and for a list that contains only empty lists.

Svar/Answer:

6.

Skrifið fall `zip2` í Scheme, CAML, Morpho eða Haskell sem tekur tvíundaraðgerð (fall) og tvo jafnlanga lista sem viðföng og skilar lista þeirra útkomna sem fást þegar tvíundaraðgerðinni er beitt á gildin í listunum, par fyrir par. Til dæmis, í Scheme þá ætti segðin `(zip2 + '(1 2 3) '(4 5 6))` að skila listanum `(5 7 9)`. Notið einungis einfaldar aðgerðir svo sem `car`, `cdr`, `cons`, `null`?

Write a function `zip2` in Scheme, CAML, Morpho or Haskell that takes as argument a binary operation (function) and two lists of equal lengths and returns the list of the results from applying the binary operation pairwise to the values in the lists. For example, in Scheme the expression `(zip2 + '(1 2 3) '(4 5 6))` should return the list `(5 7 9)`. Use only simple operations such as `car`, `cdr`, `cons` and `null`?

Svar/Answer:

7.

Skrifið ykkar eigin útgáfur af föllunum tveimur sem í CAML Light eru kölluð `it_list` og `list_it`. Í Haskell eru þau kölluð `foldl` og `foldr`. Þið megið skrifa þessi föll í Scheme, CAML, Morpho eða Haskell. Notið ekki lykkjur í Morpho. Kallið föllin `myLeft` og `myRight`. Þið megið nota aðra röð viðfanga en í `it_list` og `list_it`. Sjáið til þess að a.m.k. annað fallið sé halaendurkvæmt og tiltakið hvort það er. Notið aðeins einföld innbyggð föll svo sem `car`, `cdr` og `null?`.

Write your own versions of the two functions that in CAML Light are called `it_list` and `list_it`. In Haskell they are called `foldl` and `foldr`. You may write these functions in Scheme, CAML, Morpho or Haskell. Do not use loops in Morpho. Name the functions `myLeft` and `myRight`. You may use another ordering of argument than that used in `it_list` and `list_it`. Make sure that at least one of the functions is tail recursive and indicate which one that is. Use only simple builtin functions such as `car`, `cdr` and `null?`.

Halaendurkvæma fallið er/The tail recursive function is:

Forritstexti (með lýsingum)/Program text (with descriptions):

8.

Skrifið fall **mapreduce** í Scheme, CAML, Morpho eða Haskell þannig að (í CAML) segðin **mapreduce op f x u** sé jafngild segðinni **list_it op (map f x) u**. Notið aðeins einfaldar innbyggðar aðgerðir í lausninni, so sem `hd`, `tl`, `::` og `==`. Ekki má nota `map` eða `list_it`. Munið að `list_it` reiknar frá **hægri til vinstri** öfugt við fallið sem þið skiluðuð sem heimaverkefni.

Write a function `mapreduce` in Scheme, CAML, Morpho or Haskell so that the expression **mapreduce op f x u** is equivalent to the expression **list_it op (map f x) u**. Use only simple built-in functions in the solution, such as `hd`, `tl`, `::` and `==`. Do not use `map` or `list_it`. Remember that `list_it` computes from **right to left**, the reverse of the order in the function you programmed as an assignment.

Svar/Answer:

Part III – Modular Programming, etc.**Hluti III – Einingaforritun o.fl.**

Answer at least two questions in this part – Remember to answer at least 10 questions in total

Svarið að minnsta kosti tveimur spurningum í þessum hluta – Munið að svara a.m.k. 10 spurningum í heild

9.

Útfærið, að hluta, einingu fyrir fjölnota forgangsbiðröð í Morpho eða Java. Sýnið eftirfarandi.

Implement, partially, a module for a generic priority queue in Morpho or Java. Show the following.

- a. Hönnunarskjal sem inniheldur lýsingar (notkun/fyrir/eftir) fyrir öll innflutt og útflutt atriði einingarinnar.
A design document that contains descriptions (usage/pre/post) for all items imported to or exported from the module.
- b. Smíð einingarinnar, þar sem sleppa má útfærslu allra aðgerða nema þeirri sem fjarlægir gildi úr forgangsbiðröðinni.
Athugið að sýna þarf fastayrðingu gagna.
An implementation of the module, where you can skip implementing all operations except the operation to remove a value from the priority queue. Note that you must show a data invariant.

Unnt skal vera að nota eininguna (eða klasann í Java) til búa til forgangsbiðraðir fyrir pör (k,v) þar sem sérhvert k er er samanburðarhæft við sérhvert annað k og v er hvers konar gildi sem er, innan þeirra marka sem forritunarmálið setur. Í Morpho ráðið þið hvort forgangsbiðröðin er útfærð sem hlutur eða ekki. It should be to use the module (or the class, in Java) to make a priority queue for any pairs (k,v) of values where each k is comparable to any other k and where v is any value at all, within the constraints posed by the programming language. In Morpho you may choose whether the priority queue is implemented as an object or not.

Svar/Answer:

10.

Hverjar af eftirfarandi fullyrðingum eru í samræmi við meginregluna um upplýsingahuld? Það gætu verið núll, ein eða fleiri. Tvö röng svör gefa núll stig.

Which of the following statements are in accordance with the principle of information hiding? There might be zero, one or more. Two wrong answers give zero points.

- a. Tilgangur upplýsingahuldar er að verja iðnaðarleyndarmál.
The purpose of information hiding is to protect trade secrets.
- b. Fastayrðing gagna einingar skal ekki vera aðgengileg notendum einingarinnar.
The data invariant of a module should not be accessible to the users of the module.
- c. Fastayrðing gagna einingar skal halda leyndri fyrir smiðum einingarinnar.
The data invariant of a module should not be known to the implementors of the module.
- d. Tilgangur upplýsingahuldar er að auðvelda viðhald.
The purpose of information hiding is to make maintenance easier.
- e. Gefa skal notendum einingar fullkomnar upplýsingar um smíð einingarinnar.
The users of a module should be given perfect information about the implementation of the module.
- f. Fastayrðing gagna skal vera hluti af opinberu hönnunarskjali einingarinnar.
The data invariant should be part of the public design document for the module.
- g. Smiðir einingar geta breytt fastayrðingu gagna einingarinnar.
The implementors of a module can change the data invariant of the module.

Svar/Answer:

a	b	c	d	e	f	g

11.

Íhugið klasa A og B þar sem B er undirklasi A. Gerið ráð fyrir að klasi A innihaldi boð f með eftirfarandi lýsingu.

// Notkun: $z = t.f(x)$;

// Fyrir: $0.5 \leq x \leq 2.0$

// Eftir: $|z^2 - x| < 0.01$

Gerið ráð fyrir að í klasi B sé boðið f endurskilgreint með

// Notkun: $z = t.f(x)$;

// Fyrir: F_B

// Eftir: E_B

Consider classes A and B where B is a subclass of A. Assume that class A contains a message f with the following description.

// Usage: $z = t.f(x)$;

// Pre: $0.5 \leq x \leq 2.0$

// Post: $|z^2 - x| < 0.01$

Assume that in class B the message f is redefined with

// Usage: $z = t.f(x)$;

// Pre: F_B

// Post: E_B

Hverjir af eftirfarandi möguleikum fyrir F_B og E_B væru þá í lagi? Eitt rangt svar gefur núll fyrir dæmið.

Which of the following possibilities for F_B and E_B would then be acceptable? One wrong answer yields zero for the question.

Svar / Answer:

F_B	Í lagi / Acceptable
$0.2 \leq x \leq 1.5$	
$0.4 < x \leq 1.4$	
$0.1 \leq x \leq 5.5$	
$0.9 \leq x \leq 1.5$	

E_B	Í lagi / Acceptable
$ z^2 - x < 0.1$	
$ z^2 - x \leq 0.005$	
$ z^2 - x < 0.02$	

Part IV – Miscellaneous**Hluti IV – Ýmislegt**

Answer at least one question in this part – Remember to answer at least 10 questions in total

**Svarið að minnsta kosti einni spurningu í þessum hluta –
Munið að svara a.m.k. 10 spurningum í heild**

12.

Lýsið ruslasöfnunaraðferðinni sem gengur undir nafninu **merkja og sópa**. Koma þarf fram hvað gerist í merkja fasanum, hvað eru rætur, hvað gerist í sópa fasanum, hvað frjálsi listinn er og undir hvaða kringumstæðum ruslasöfnunin fer af stað.

Describe the garbage collection method called **mark and sweep**. The things to describe include what happens during the mark phase, what are roots, what happens in the sweep phase, what the free list is and under what circumstances the garbage collection is initiated.

Svar/Answer:

13.

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/Answer:

(auð blaðsíða/empty page)

(auð blaðsíða/empty page)

(auð blaðsíða/empty page)

(auð blaðsíða/empty page)