

הוראות והנחיות

1. **משך:** משך המבחן שלוש שעות, לא תינתנה הארכות.
2. **חומר עזר:** אין.
3. **מבנה:** במבחן עשר שאלות. לכל השאלות משקל שווה, אך קשיין של השאלות אינו שווה. כל שאלה מתחילה בעמוד חדש. במבחן 14 עמודים.
4. **מקום:** יש לענות על מחברת המבחן בלבד. בד"כ החלל אשר הותרנו במחברת בעבור התשובה גדול בהרבה מנפח התשובה אשר לה אנו מצפים.
5. **מחברות טיוטה:** מותר להשתמש במחברות טיוטה, אך אין להגיש אותן.

נא לכתוב בתמציתיות, בכתב ברור, בעט, ולהשתמש במונחים מדוייקים. כתיבה בעיפרון, בעט מחיק, בכתב לא ברור, כמו גם התנסחות במונחים לא מדוייקים עלולה לפגוע בציון.

שאלה 1 (ניתוח תכנית)

נתון קטע קוד הכתוב הכתובה בשפת תכנות אשר אולי נלמדה ואולי לא נלמדה בקורס. עיין בתכנית, והסק מסקנות ביחס לשפה על פי השאלות הבאות. בשאלות בהן נתבקשת לנמק, נמק בתמצית. אין טעם לנחש בשאלות אלו, משום שתשובות ללא נימוקים לא תזכנה בציון. (שים לב לכך שלא כל המידע המופיע בתכנית, יופיע בשאלות עליה.)

```
1. #!/usr/bin/dmd -run
2. /* sh style script syntax is supported! */
3. /* Hello World in D
4.     To compile:
5.         dmd hello.d
6.     or to optimize:
7.         dmd -O -inline -release hello.d
8.     or to get generated documentation:
9.         dmd hello.d -D
10. */
11. import std.stdio; // References to commonly used I/O routines.
12. void main(char[][] args) { // 'void' here means return 0 by default.
13.     // Write-Formatted-Line
14.     writeln("Hello World, " // automatic concatenation of string literals
15.         "Reloaded");
16.     // Strings are denoted as a dynamic array of chars 'char[]'
17.     // auto type inference and built-in foreach
18.     foreach(argc, argv; args) {
19.         // OOP is supported, of course! And automatic type inference.
20.         auto cl = new CmdLin(argc, argv);
21.         // 'writeln' is the improved 'printf' !!
22.         // user-defined class properties.
23.         writeln(cl.argnum, cl.suffix, " arg: %s", cl.argv);
24.         // Garbage Collection or explicit memory management - your choice!!!
25.         delete cl;
26.     }
27.     // Nested ...
28.     struct specs {
29.         // all vars. automatically initialized
30.         int count, allocated;
31.     }
32.
33.     // Note that declarations read right-to-left.
34.     // So that 'char[][]' reads as an array of an array of chars.
35.
36.     specs argspecs(char[][] args)
37.     // Optional (built-in) function contracts.
38.     in {
39.         assert (args.length > 0); // assert built in
40.     } out(result){
41.         assert(result.count == CmdLin.total);
42.         assert(result.allocated > 0);
43.     }
44.     body {
45.         specs* s = new specs;
46.         // no need for '->'
47.         s.count = args.length; // The 'length' property is number of elements.
48.         s.allocated = sizeof(args).sizeof; // built-in properties for native types
49.         foreach(argv; args)
50.             s.allocated += argv.length * sizeof(argv[0]);
51.         return *s;
52.     }
53.
```

```
54. // built-in string and common string operations, e.g. '~' is concatenate.
55. char[] argcmgs = "argc = %d";
56. char[] allocmsg = "allocated = %d";
57. writefln(argcmgs ~ ", " ~ allocmsg,
58.         argspecs(args).count, argspecs(args).allocated);
59. }
```

א. לאיזו פרדיגמה שייכת השפה על פי הדוגמה? נמק.

ב. האם על פי הדוגמה השפה ספראטיסטית, טרמינסטית, או אולי משהו אחר? נמק.

ג. באיזו שיטה (או שיטות) על פי הדוגמה משתמשת השפה לציון הערות? אין לנמק.

ד. האם על פי הדוגמה בשפה נהוגה טיפוסיות סטאטית או דינאמית? נמק.

ה. מהם שני טיפוסים המילולונים המופיעים בדוגמה?

ו. מהו סוג הלולאה המופיע בדוגמה? נמק!

ז. מהם כל המילים השמורות המופיעות בדוגמה. אין לנמק.

ח. אילו מבין מילים שמורות אלו הם מזהים שמורים? אין לנמק.

ט. באלו בנאי פקודות עושה שימוש הדוגמה? אין לנמק.

י. באילו שורה/שורות מופיעות פקודות אטומיות בדוגמה? אין לנמק.

שאלה 2 (מיני ליספ)

בשאלה זו, עליך להשתמש בדקדוק של ליספ, ובפונקציות atom ו car, cdr, cons, cond, eq, defun. אין להשתמש במספרים בליספ, אלא ברשימות ובביטויי S בלבד.

א. כתוב את הפונקציה SNOC אשר מקבלת שני פרמטרים: ביטויי S אשר שמו יהיה X ורשימה אשר שמה יהיה XS ומחזירה את הרשימה המתקבלת מהוספת X בסוף הרשימה XS. סדר הפרמטרים ל- SNOC יהיה הפוך מזה של הפונקציה CONS. ניתן, אך לא חובה, להגדיר פונקציות עזר.

פונקצית עזר I	הפונקציה SNOC
פונקצית עזר II	פונקצית עזר III

ב. כתוב את הפונקציה LISTABLE אשר מקבלת ביטויי S ומחזירה את הערך המקובל בליספ ל-TRUE אם הביטוי ניתן להכתב בכתיב הרשימות (כלומר כרשימה של אטומים או של רשימות), ואת הערך המקובל בליספ ל-FALSE אם הביטוי אינו יכול להכתב בכתיב זה.

פונקצית עזר I	הפונקציה LISTABLE
פונקצית עזר II	פונקצית עזר III

שאלה 3: (STORAGE)

א. מה פשר ראשי התיבות RTTI

ב. האם RTTI נחוצ ל-STATIC TYPING, ל-DYNAMIC TYPING או לשניהם?

ג. תאר את האלגוריתם MARK-AND-SWEEP

ד. הסבר האם האלגוריתם דורש RTTI ומדוע, או שאינו צריך RTTI ומדוע

ה. רשום את השלבים במחזור החיים של תא זיכרון במודל הזיכרון האוטופי. נניח שקיים RTTI. באיזה מבין השלבים הללו, נוצר ה RTTI?

שאלה 4 (TYPES)

פתור את המשוואות הבאות

1. $X = (X+1)*X$

2. $X = \text{Int} * X$

3. $X = 1 + X$

4. $X = 1 + \text{Int} * X + \text{Int} * X * X$

5. $X = \text{Int} * (1 + Y); Y = \text{Real} * (1 + X) * (1 + X)$

שאלה 5 (Abstraction)

א. באר את המשפט הבא: כידוע, חישוב נורמלי, הוא לא נורמלי, שכן באורח נורמלי, החישוב הלא-נורמלי, כלומר, הנוקשה, הוא זה שנהוג.

ב. נתון הציטוט הבא מויקיפדיה:

In certain computer programming languages, the Elvis operator, often written `?:`, or `or` `||`, is a binary operator that returns its first operand if that operand evaluates to a true value, and otherwise evaluates and returns its second operand. The Elvis operator is a variant of the ternary conditional operator, `? :` in the sense that the expression with the Elvis operator `A ?: B` is approximately equivalent to the expression with the ternary operator `A ? A : B`. In a language that supports the Elvis operator,

`x = f() ?: g()`

is equivalent to (using the conditional ternary operator)

`x = f() ? f() : g()`

except that it does not evaluate the `f()` twice if it is true.

בהנחה שיש בשפת C++ מילה שמורה `normal` המציינת העברת פרמטרים בשיטה זו, כתוב פונקציה בשם `elvis` המממשת אופרטור זה

שאלה 6 (Intro and More)

(2) א. מהם NAMEABLES?

(3) ב. מצא בשפת פרולוג קבוצה a של ישויות, שהשמות שלהם יכולה להילקח רק מקבוצה A של שמות, וקבוצה b של ישויות שהשמות שלהם יכולים להילקח רק מקבוצה B של שמות, והקבוצות A ו B זרות.

(3) ג. נתונים מספר סוגים של NAMEABLES של שפת פסקל. תן דוגמה בשפת תכנית לשימוש בכל אחד מהם
1. נקודות בתכנית, אליהן אפשר לבצע GOTO.

2. קבועים.

3. טיפוסים

4. משתנים

5. פונקציות

6. פרצדורות

(2) ד. איזה סוג של NAMEABLES בפסקל שונה מכל השאר (במובן של סעיף ב)? הסבר.

שאלה 7: ML

נבילה התקינה את המפרש של ML המוכר לך, אך גילתה שצדד חיבל בגירסא זו. לאחר מחקר, גילתה נבילה מה עשה אותו צדד. הזד השמיט את המימוש של האופרטורים andalso ו orelse. מלבד חבלה זו הגירסא זהה לחלוטין לגירסא המוכרת לכם. **נבילה** לא נואשה, והחליטה להוסיף במו ידיה מימוש של האופרטורים הללו וכתבה את המימוש הבא:

```
infix andalso;
```

```
infix orelse;
```

```
fun x andalso y = if x then y else false;
```

```
fun x orelse y = if x then true else y;
```

מוצטפא טוען כי ההתנהגות של האופרטורים במימוש של זהבה שונה מההתנהגות הרגילה ב ML. בכדי להוכיח את טענתו של **מוצטפא**, כתוב קטע קוד (קצר ככל הניתן) שבו ההתנהגות תהיה שונה אם נשתמש בגירסא של זהבה לעומת הגירסא הרגילה של ML. כתוב מה תהיה התוצאה בכל אחד מהמקרים

ממש את האופרטורים הנ"ל כך שההתנהגות תהיה זהה להתנהגות המצופה ב ML, או הסבר מדוע לא ניתן לעשות זאת.

שאלה 8: ML

נתונות ההצהרות הבאות בשפת ML. בכל מקום בו השארנו לכם מקום לכתובה יש לציין את תגובת המפרש להצהרה האחרונה. אם יש אזהרות או שגיאות, יש לציין זאת במפורש. אין צורך לרשום את השגיאה המדויקת שהמפרש יפיק, מספיק לתאר אותה במילים.

```
fun f1 a b = f1 a b
```

```
fun f2 (x::xs) [y::ys] = y::x;
```

```
fun f3 x y z = (x,x(y),y(z));
```

```
fun f4 f = f f4;
```

```
fun f5 (f5::_ ,0,_) = f5+1 | f5 ([],3,f5) = f5(1);
```

שאלה 9: ML

נתונות ההגדרות הבאות של hd ו-tl:

```
exception EmptyList;
fun hd [] = raise EmptyList | hd (x::xs) = x;
fun tl [] = raise EmptyList | tl (x::xs) = xs;
```

השתמשו בהגדרות הנ"ל בשביל לכתוב את הפונקציות הבאות ללא שימוש בביטויי תנאי או pattern matching (פרט ל-handle EmptyList).
א. הפונקציה

append(lst1, lst2)
המקבלת שתי רשימות lst1, lst2 ומחזירה את השרשור של lst2 אחרי- lst1.

ב. הפונקציה

length_inner (lst, n)
המקבלת רשימה ומחזירה את האורך שלה. חובה להשתמש ברקורסיית זנב!

ג. איך תראה הפונקציה length שתעטוף את length_inner? השלימו:

```
fun length lst = length_inner(_____, _____);
```

שאלה 10: Prolog

בשאלה זו, ננסה לממש את מיני-ליספ באמצעות פרולוג.

לשם כך, נשתמש בפרדיקט דו מקומי, ששמו s , אשר יציין ביטוי s ובאטום פרולוג מיוחד (פרדיקט 0 מקומי) ששמו n , אשר יציין את האטום nil של ליספ. נניח שיש גם פרדיקט חד מקומי בפרולוג ששמו $atom$ אשר מצליח אם הארגומנט שלו הוא אטום של פרולוג. כתוב את הפרדיקטים הבאים בפרולוג (ניתן להשתמש בפרדיקטי עזר, ובכל סעיף ניתן להשתמש בפתרונות של כל הסעיפים האחרים)

A. $null(X)$ that holds whenever $(null\ X)$ in lisp holds

B. $eq(X,Y)$ that holds whenever $(eq\ X\ Y)$ in lisp holds

C. $cons(X,Y,Z)$ that holds whenever in lisp $(cons\ X\ Y)$ returns Z

D. $rev(X,Y)$ that holds whenever X is the same list as Y but in reverse order

E. $listable(X)$ that holds whenever X can be presented in lisp using lists, lists of lists, etc.
