

## שפות תכנות -- מבחן לדוגמה, סמסטר א', 2022/2023

- יש לענות אך ורק בטופס המבחן. מחברת הבחינה לא תיבדק.
- המחלקה למדעי המחשב
- מרצה: ד"ר יוני זוהר
- מתרגל: צביקה ברגר
- חומר עזר חיצוני: אין
- משך המבחן: 180 דקות
- מומלץ להשתמש במחברת הבחינה כדי לכתוב טיוטה של התשובות, ואז להעתיקן למקום המתאים בטופס.
- במידת הצורך, בסוף הטופס ישנם דפים נוספים בהם ניתן לכתוב תשובות לשאלות.
- בהצלחה!
- שוב: יש לענות אך ורק בטופס המבחן. מחברת הבחינה לא תיבדק.

1. להלן ריצה אינטראקטיבית של *utop*, האינטרפרטר של *OCaml*. שורה שמתחילה ב"*utop #*" היא שורת קלט מהמשתמש. שורה ללא *utop #* היא התשובה של *utop*. מהריצה הושמטו חלקים, המסומנים בקו. השלימו אותם. אם מתקבלת שגיאה, אין צורך לרשום אותה במדויק, ומספיק לרשום *error*.

(א)

```
utop # let curry = (fun f -> (fun x -> (fun y -> (f (x, y)))));;
```

```
val curry : _____ = <fun>
```

(ב)

```
utop # let third = (fun (x,y,z)->z);;
```

```
_____
```

(ג)

```
utop # (third(third, third, third))(third, third, 3);;
```

```
_____
```

(ד)

```
utop # third 3;;
```

```
_____
```

(ה)

```
utop # third(1, 2, third)(third, third, third)(1,2,3);;
```

```
_____
```

2. להלן קטע קוד באוקמל שחלקו הושמט.

```
let rec fold_right = fun f l init -> match l with  
  [] -> init  
  | h::t -> f h (fold_right f t init);;
```

let f = \_\_\_\_\_;;

let init = \_\_\_\_\_;;

let maxi = fun ell -> fold\_right f ell init;;

(א) השלימו את ההגדרות של  $f$  ו- $init$  כך ש- $maxi$  תהיה פונקציה שמקבלת רשימה של מספרים חיוביים ומחזירה את האיבר המקסימלי שלה, אם אינה ריקה. אם הרשימה ריקה, האיבר המקסימלי מוגדר להיות  $-1$ .

(ב) מה הטיפוס של  $fold\_right$ ?

```
type tree = Nil | N of tree*tree;;  
let rec ni t = match t with | Nil -> 1 | N(t1, t2) -> (ni t1) + (ni t2) ;;  
let rec no t = match t with | Nil -> 0 | N(t1, t2) -> (no t1) + (no t2) + 1;;
```

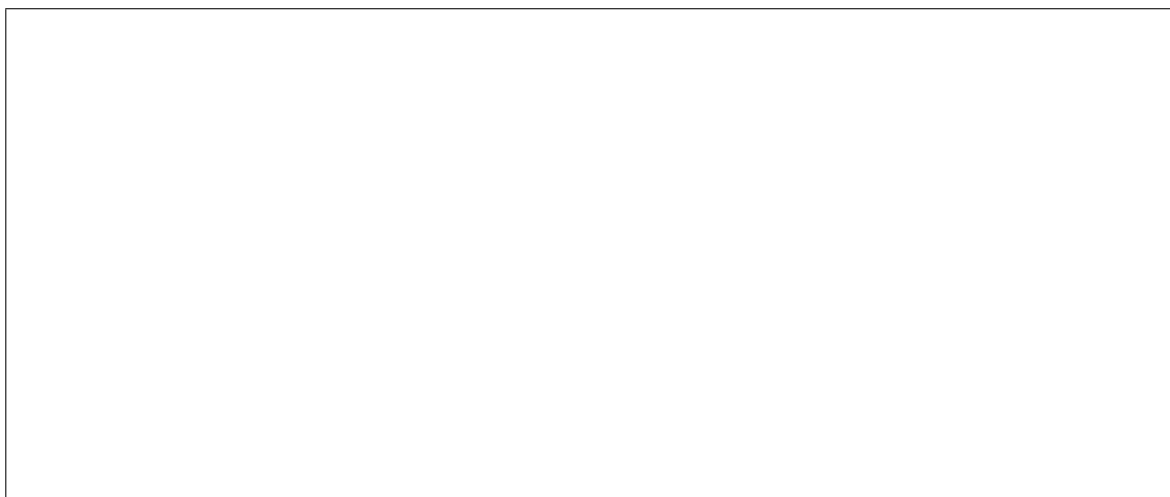
הוכיחו באינדוקציה כי  $ni(\ell) = no(\ell) + 1$ .



4. יהי  $s_0$  מצב כך ש- $s_0x = 5$  ו- $s_0y = 5$ . הוכיחו כי:  $\langle while\ x \leq y\ do\ x := x + 1, s_0 \rangle \Rightarrow^* s_0 [x \mapsto 6]$  על ידי שימוש בכללי ההיסק של הסמנטיקה המבנית.

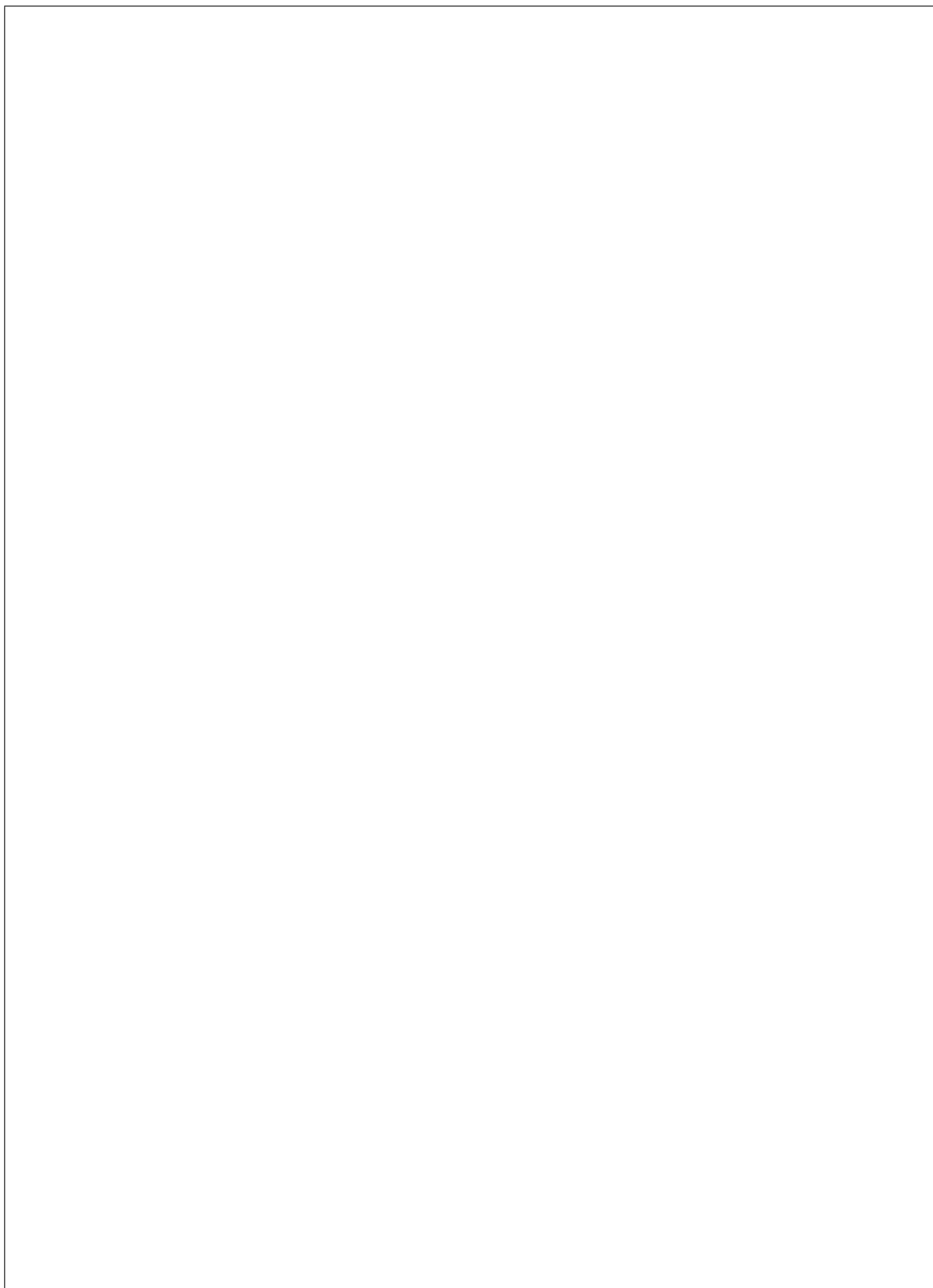


5. הגדירו קומבינטור  $pow$  שיקיים לכל  $n$  ו- $m$  טבעיים:  $(pow\ c_n\ c_m) = c_{n^m}$ . מותר להשתמש בקומבינטורים מדף הנוסחאות מבלי להגדיר מחדש. כל קומבינטור אחר שתצו להשתמש בו, יש להגדיר במפורש.



6. הפעילו רדוקציות  $\beta$  על הביטוי הבא עד שתגיעו לביטוי שממנו לא ניתן לעשות רדוקציית  $\beta$ . בצעו את הרדוקציות באיזה סדר שתבחרו.

$$((\lambda f. ((\lambda g. ((f f) g)) (\lambda h. (k h)))) (\lambda x. (\lambda y. y)))$$



(א) הוכיחו על ידי שימוש בכללי ההטפסה:  $\vdash (\lambda x : Bool. if\ x\ then\ true\ else\ x)\ true : Bool$

(ב) מצאו  $T$  כך ש- $\vdash (f\ x)\ y : Bool$   $x : Bool, y : Bool, f : T$



(א) רשמו את המשוואות המתאימות לבעיית הסקת הטיפוסים של הביטוי הבא:

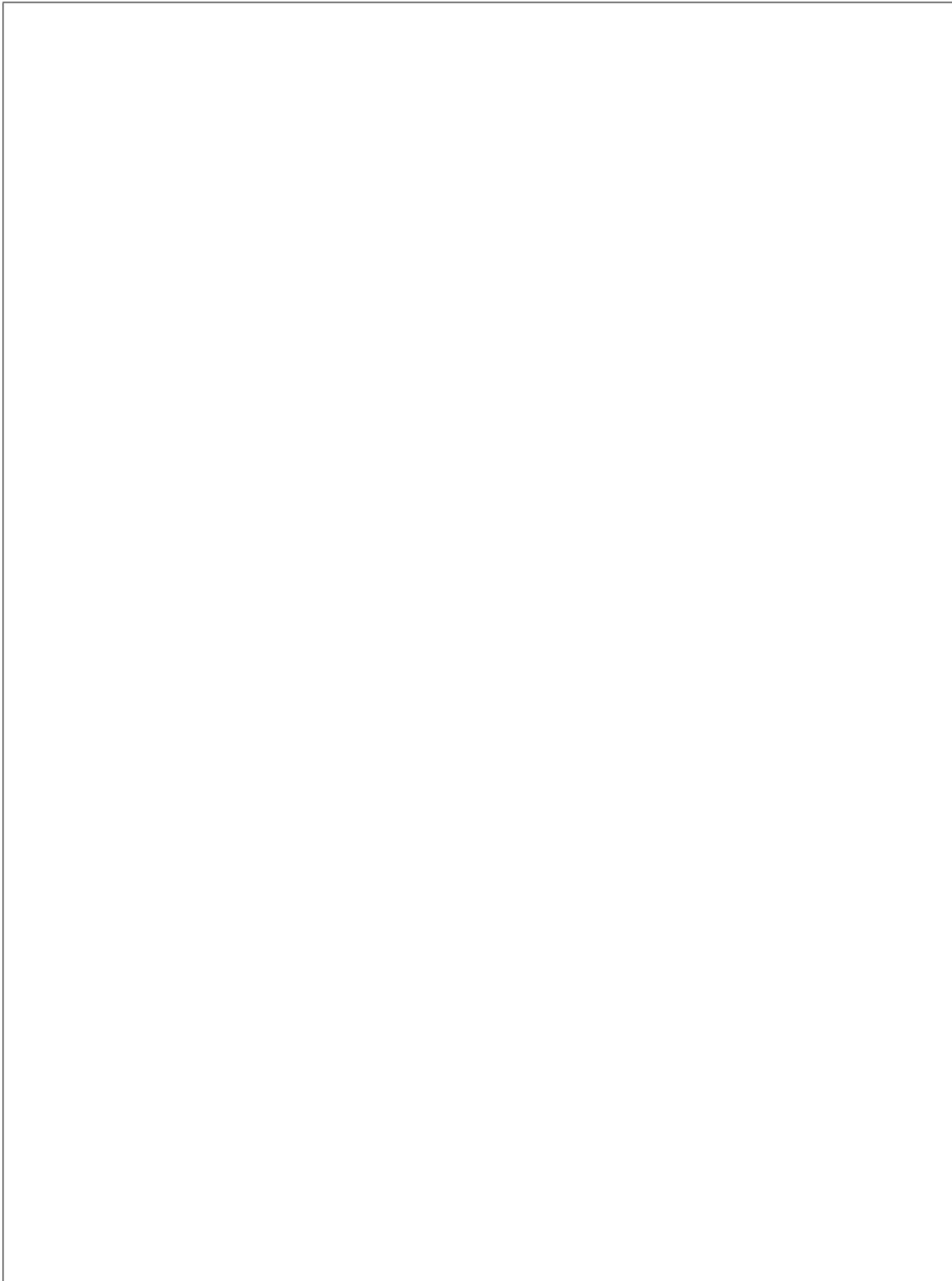
$$\text{fun } x \rightarrow (\text{fun } y \rightarrow (\text{fun } z \rightarrow ((x\ y)\ z)))$$

(ב) הציגו החלפה  $\sigma$  שמאחדת את המשוואות מהסעיף הקודם.

(ג) מהו הטיפוס של הביטוי  $\text{fun } x \rightarrow (\text{fun } y \rightarrow (\text{fun } z \rightarrow ((x\ y)\ z)))$ ? מותר להשתמש בסעיפים קודמים אך אין חובה לעשות זאת.

9. כתבו תכנית פרולוג בשם *dupli* שמשכפלת את כל אחד מהאיברים של רשימה. להלן ריצה לדוגמה:

?- dupli([a,b,c,c,d],X).  
X = [a,a,b,b,c,c,c,c,d,d]



10. להלן קוד בג'אווה סקריפט עם חלקים שהושמטו, שמטרתו היא לממש את הפונקציה שמחזירה לכל מספר טבעי  $n \geq 1$  את סכום המספרים מ-1 עד  $n$ , עם *memoization*. השלימו את המקומות החסרים. ניתן לכתוב מספר פקודות באותה השורה, אך במקרה כזה יש להפרידן ב-  
;

```
let cache = [];
```

```
cache[1] = _____
```

```
function memoizeSum(n) {
```

```
  if (cache[n]) {
```

```
    _____
```

```
  } else {
```

```
    _____
```

```
  }
```

```
}
```







