

Esame di Linguaggi e Paradigmi di Programmazione (6 CFU)

FAC SIMILE

Regolamento

- Il tempo a disposizione per la risoluzione degli esercizi di laboratorio è di **20 minuti**.
- Non è consentita la consultazione di appunti, dispense, libri o l'uso di laptop, tablet, ecc.
- Per la risoluzione degli esercizi 1 e 2 è possibile usare un editor a scelta, l'ambiente interattivo GHCi e Hoogle sulla postazione di laboratorio occupata.
- Il tempo a disposizione per la risoluzione degli esercizi 3 e 4 è di **40 minuti**.

1 Laboratorio

Esercizio 1 (7 punti). *Definire una funzione che, applicata a una lista `xs`, ritorna la sotto-lista contenente tutti e soli gli elementi di `xs` **in posizione pari**, nello stesso ordine in cui compaiono in `xs` e assumendo che il primo elemento della lista si trovi in posizione 0. È vietato fare uso di funzioni della libreria standard ad eccezione di `mod` e quelle che hanno un nome simbolico, come `+`, `.`, ecc.*

Esercizio 2 (7 punti). *Ripetere l'esercizio precedente, questa volta senza fare uso esplicito della ricorsione ma potendo usare tutte le funzioni definite nel modulo `Prelude`.*

2 Teoria

Esercizio 3 (8 punti). Applicare l'algoritmo di inferenza all'espressione

```
(\f g x → f x (g x)) (·)
```

per determinarne, se esiste, il tipo più generale.

Esercizio 4 (8 punti). Dimostrare la proprietà

```
length . foldr (·) [] = foldr (const (1 +)) 0
```

dove

```
const :: a → b → a  
const x _ = x
```

Indicare i principi di dimostrazione applicati e giustificare ogni passaggio della dimostrazione:

- proprietà note delle operazioni aritmetiche (es. commutatività e associatività di $+$ e $*$) possono essere assunte ma vanno comunque **menzionate**;
- eventuali riferimenti a funzioni di libreria (es. con `foldr.1`) vanno accompagnati dalla definizione **completa** della funzione (es. di `foldr`);
- eventuali altre proprietà utilizzate vanno **dimostrare** esplicitamente.