

Name	Vorname	Matrikelnummer	Punkte

Verschiedene semantische Bereiche werden durch verschiedene Typkonstruktoren abgebildet sowie Operationen in diesen Bereichen durch **Monad**-Instanzen.

Definieren Sie die Bereiche (Typen) und geben Sie die Implementierungen von **return** und **bind (>>=)** an.

/3 P

- Aktionen, die eine Speicherbelegung ändern

– Typ: Action a = ...

– return x = ...

– m >>= f = ...

/2 P

- Verarbeitung von Continuations

– Typ: CPS a = ...

– return x = ...

– (Zusatz) m >>= f = ...

/2 P

- Parser mit Tokentyp c

– Typ: Parser c a = ...

– return x = ...

– (Zusatz) m >>= f = ...

/3 P

- Rechnungen, die möglicherweise fehlschlagen

– Typ: data Maybe a = ...

– return x = ...

– m >>= f = ...

Name	Vorname	Matrikelnummer	Punkte

Im folgenden Programmtext soll eine Funktion $f :: \text{Nat} \rightarrow \text{Nat}$ durch Rekursion definiert werden.

```
letrec { f = \ x -> if x > 5 then x - 1 else f (f (x+3)) } in f 0
```

Die Bedeutung von f kann als Limes einer aufsteigenden Folge

$$f_0 = \perp, f_1 = F(f_0), f_2 = F(f_1), \dots$$

von partiellen Funktionen erhalten werden.

Geben Sie den Operator F und seinen Typ an.

/3 P

Geben Sie f_0, f_1, f_2, f_3 an.

/3 P

Wie kann man Rekursion mittels Fixpunktkombinator ODER mittels Speicherzustandsänderung realisieren?

/4 P

Name	Vorname	Matrikelnummer	Punkte

Bestimmen Sie den allgemeinsten Unifikator der Typ-Ausdrücke

$$(t_2 \rightarrow (t_1 \rightarrow (t_0 \rightarrow t_3))) \quad \text{und} \quad (((\text{Int} \rightarrow t_0) \rightarrow t_1) \rightarrow t_2)$$

/4 P

Begründen Sie, daß das folgende Programm nicht einfach typisierbar ist:

```
let { id = \ x -> x }  
in let { c = \ x -> \ y -> y x }  
    in c c id 3 id
```

/3 P

Das Programm ist polymorph typisierbar. Fügen Sie alle dafür nötigen Typ-Deklarationen, Typ-Abstraktionen und -Argumente ein.

/3 P

Name	Vorname	Matrikelnummer	Punkte

Für die Phasen des in der Vorlesung behandelten Compilers: Welche syntaktischen Einschränkungen gelten jeweils für das bei der Transformation erzeugte Programm?

- CPS-Transformation

/2 P

- Closure Conversion

/2 P

- Lifting

/2 P

- Register-Zuweisung

/2 P

Welche Eigenschaft der Quellsprache führt dazu, daß bei der Ausführung des Zielprogramms ein Garbage Collector benötigt wird?

/2 P