Assignment Project Exam Help Lecture 5. Functions Help

https://powcoder.com

Overview

Assignment Project Exam Help

- lambda expression
- recursion
- high-order functions

 Lilp Sungtion Det Wcoder.com
- Syntax
- Segnantics WeChat powcoder

Abstraction in Programming Languages

Assignmente Porojectus Exam Help fixed abstraction – you cannot change computation

representing values

https://poweoder.com

► Function (procedure, method):

parameterization for computation

you a reuse the fluctional in for different cardet in the CI

language features that can define a procedure and call a procedure

Lambda Expression

Defining anonymous function

Assignment/Bit by the function of defining function to defining function to the function of defining function to the function of the function

- Formpare to ALGOL family languages: G, C++, Java ... (syntax):
 - formal parameter name only, no types precede or follow
 - no explicit return is needed
- Compare to ALGOL family languages: C, C++, Java ... (semantics): Protefurs and methods: propy of the D aid A a legical of code
 - adjust the environment
 - jump to the location

Lambda abstraction:

- generate runtime values
- each of the runtime values can be used multiple times

```
lambda //Lambda special form for defining functions
(x) //List of formal parameter names of the function
(+ https://powcoder.com
```

Examples: Calling the Lambda function

```
Assignment Project Exam Help
//Operator: function being called
//Operands: list of actual parameters
https://powcoder.com

Add Wechat powcoder
```

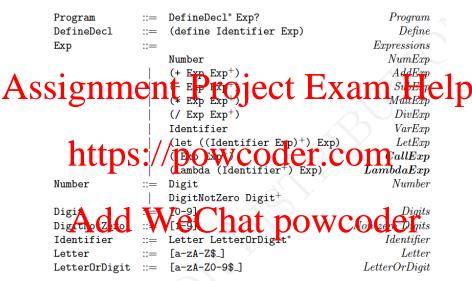


Figure 5.1: Grammar for the Funciang Language. Non-terminals that are not defined in this grammar are exactly the same as that in Definelang.

Examples: Combine with Let and Define

Assignment Project Exam Help ((identity (lambda (x) x))) //Naming the function (identity 1) //Function call) https://powcoder.com

```
$\frac{\( \text{define square (lambda } (\times \text{ (\text{x x x})))}}{\text{Add.}(\text{4})}\)

Add.(\text{4})
```

Writtpre-Lambdowesiondwith Leconnicine

- ▶ if expression: three mandatory expressions the condition, then, and entypesons//powcoder.com

 comparison expression: >, <, =

Control Structure: Grammar



Figure 5.6: Extended Grammar for the Funciang Language. Non-terminals that are not defined in this grammar are same as that in figure 5.1.

- 1. pair: 2 tuple (fst, snd)
- 2. list: empty list, or 2 tuple
- 3. https://poweoder.com

 $List := (list) \mid (cons \, \mathtt{val} \, List), \, \mathtt{where} \, \mathtt{val} \, \in \, \mathtt{Value}$

List and its built-in functions in FuncLang

Assignment groject Exam Help Null?: check if a list is a null, returns #t if that argument is an

- null?: check if a list is a null, returns #t if that argument is an emptylist else return #f
- chr: get the first element of a pair, e.g. (car (list 11 1))
- cdr. get the second element of a pair, e.g., (cdr (fist 342, 831, 327))
- cons:
 - Constructing a pair, e.g., (cons 2 3) (cons 541 (list 342))

 Alists can also be constructed by using the cons keyword as is shown (construction)

 (1)

Examples: list with functions

```
Assignment Project Exam Help
    https://powcoder.com
         (define caddr
    Add WeChat powcoder
```

Examples: list and its built-in functions in FuncLang

```
Assignment Project Exam Help
 $ (cons 1 2)
 $ (contact types)://powcoder.com
 $ (define L (list 1 2 3))
 $ (car L)
 1 (cdr Add WeChat powcoder
 (23)
```

Examples: list and its built-in functions in FuncLang

```
Ssirginment Project Exam Help funciang. Value Null cannot be cast to funciang. Value spair Value spair
                        $ (cdr (list))
                     funclang. Value $\text{Null cannot be cast to funclang. Value $\text{pairVal} $ (\text{list ps://powcoder.com} $ (\text{cdr}(\text{fist 1}) \text{ps://powcoder.com} $ (\text{cdr}(\text{fist 1}) \text{ps://powcoder.com} $ (\text{cdr}(\text{fist 2}) \text{powcoder.com} $ (\text{cdr}(\text{fist 2}) \text{cdr}(\text{cdr}(\text{fist 2}) \text{cdr}(\text{cdr}(\text{fist 2}) \text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\text{cdr}(\te
                        $ (car (list 1))
                     1 (nul Ald)d WeChat powcoder
```

Grammar with List

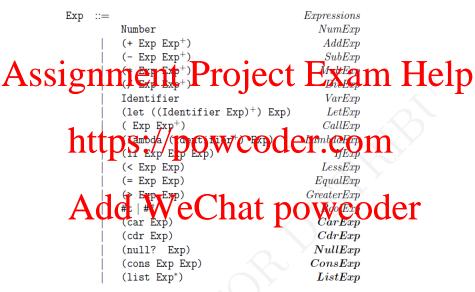


Figure 5.8: Extended Grammar for the Funclang Language. Non-terminals that are not defined in this grammar are same as that in figure 5.1.

Exercise: functions, list and control structure

Assignment Project Exam Help

With the struction of the course of the cour

```
Examples: all together
   $(if (null? I) (car I) (cdr I))
   $ (cdr (cdr (list 1 2 3)))
   (3)
   signment Project Exam Help
   $ (cdr (cdr (list 1 2 3))))
   $ (cdr (list))
   funcian Villa Sull can po W Code Value Talue Talue
   $ (car (list))
   funclang. Value $Null cannot be cast to funclang. Value $Pair Val
   $ (cdr (cdr (cdr (list 1 2 3))))
   $ (cons 3 (list))
   (3)
   (define f (lambda (x) (* x x)))
   $ (list 1 2 f)
   (1\ 2\ (lambda\ (\times)\ (*\times\times)))
   $ (list 1 2 (f 5))
   (1225)
                                           4□ > 4□ > 4 = > 4 = > = 90
```

Recursive Function

Recursive function mirror the definition of the input data type

Assignment Project Exam Help

```
httpasiboa/lip@wcoder.com

(if (null? lst1) lst2
(if (null? lst2) lst1

Add) WeChat powcoder

)
```

Define a function sum that sums the number 1 to n.

Assignment Project Exam Help \$ (define sum (lambda (n) (if (= n 1) 1 (+ n (sum (- n 1)))))) \$ (sum 1) \$ (surhttps://powcoder.com) \$ (sum 3) Add WeChat powcoder

FuncLang Programming Exercise

Assignment Project Exam Help

- Define a function that computes the factorial of a given integer n
- ► Define arrhymotion sums makes that takes the integers as a parameter and computes the sum of square of numbers from the first number to the second number.

High Order Function - take function as an input

a function that accepts a function as an argument or return a function as value

Assignment Project Exam Help

```
(define applyonone
(lambda (f) (f 1)))/powcoder.com
$(addthree applyonone) //error
$ (add Aree day lyon add three) at powcoder
(define addtwovalues (lambda (x y) (+ x y)))
$ (applyonone addtwovalues) //error
(define applyonetwo (lambda (f) (f 1 2))
$ (applyonetwo addtwovalues)
```

```
High Order Function - return a function
```

```
(lambda
    (c)
    (lambda (x) c)
Assignment Project Exam Help
   ( (lambda
    (c)
    "https://powcoder.com
   (((larAdd WeChat powcoder
    (lambda (x) c)
```

High Order Function - using function to represent data structure and its operations

```
Assemble Project Exam Help

(lambda (op)
    (if op fst snd)

) https://powcoder.com
)
(define apair (pair 3 4))
(define first (lambda (p) (p #t)))
$ (first apair) We Chat powcoder
```

- what is apair?
- what is first?

High order function: problem solving

parameterize functions: defining reusable algorithmic structures, e.g., a higher-order function that accepts an operation and a list and Sapling in period of each operation the list and the list are the list and the list are t

- identify: what is high order function (given by the problem)? what is op used as paramaterized algorithms? what are the parameters of high order function that op will apply to? op will perform on which data is ricouse and in the parameters? (elimin of the problem)
- high order function will (repeatedly) apply op on its other parameters
- ► if the high order function is recursive, what is the initial condition, and what is the supproblem of n-1.
- high order function for data structures paramyters are members, the operators, .e.g, getfirst, getsecond, on the data structure are op: a constructor for creating pairs, an observer for getting the first element of the pair, and another observer for getting the second element of the pair.

Exercise: High Order Function

Define a function filter with the signature: (define filter (lambda (test op lst) ...)) The function takes two inputs, an operator test op that should be a single argument function that returns a bodean, and lst that should be a felements. The function of that callist containing all the elements of "lst" for which the test op function returned #t.

```
$ (define gt5? (lambda/(x) (if (> x 5) #t #f)))
$ (file ttp St.)) / powcoder.com
()
$ (filter gt5? (list 1))
()
$ (filter gt5? (list 1))
(67)
$ (filter gt5? (list 1 6 2 7))
(67)
$ (filter gt5? (list 1 6 2 7 5 9))
(679)
```

- ► (define gt5? (lambda (x) (if (> x 5) #t #f)))
- (leftine filter (lambda (op I) (if (null? I) (list) (if (op (car I))) (cons (lat U) (liter op (ed D)) (twer op (ed D)). COM
- $\begin{array}{c} \begin{tabular}{lll} \begin{tabular}{llll} \begin{tabular}{lll}$

Currying

```
[the term Currying is from Haskell Curry] Model multiple argument

Sanida apprendiction in a community of the land Help
 (define plus
 (lathbda(xx))(± //)))owcoder.com
   (lambda (x)
    (lambda (y)) WeChat powcoder
```

Revisit Syntax

What is new? Funclang - Functions

ssignment Project Exam Help

- Function with a name
- High order function, including currying

 Intro 8-in/function we coder.com

 - list

- if cond truestmt falsestmt
 - ▶ #t, #f
 - < Exp Exp</p>
 - ► = Exp Exp
 - < Exp Exp</p>

Language design decisions for functions

▶ Do we require a function name? or do we allow anonymous functions? (first-class function functions are variables of the

ssignment, Project Exam Help

- Do we allow to write a function in the function body (nested function)?
- Do verallow chigh orthogonomics? (Consider C functionspinters)
 Do we allow default values in the parameters?
- Do we support variant parameters? foo(int x, ...)
- Anyalternative syntax for Chillexp: (Variable p EVV) e Chat powcoder Should we perform syntactic or semantic checks to report an invalid expression (1 1)?
- ▶ There are errors that we cannot use CFG to check but only can check them in evaluators, e.g., checking the numbers of formal paramaters and actual parameters must be equal for a CallExp.

- Any new types of values to be added?

 > sattps://powcoder.com
- ► How to implement it?

New Values for FuncLang

Assignments function if exalus Enxantassed lepparameters, return from a function and stored in the environment

```
https://powcoder.co.lylles
| Funval | Funval n | NumVal | Funval | Funval | NumVal | Funval |
```

New Values for FuncLang: Implementation

```
class FunVal implements Value {
                                                                                                                                            private Env. env:
Assignment of the Control of the Con
                                                                                                                                                        _{env} = env;
                                                                                                                                                         _formals = formals;
                                                                                                                                                        _{bodv} = bodv:
                                                                                                                                                                                 Mpowcoder.com
                                                                                                             11 public List string> formals() { return _formals; }
                                                                                                                                            public Exp body() { return _body; }
                                                                                                         13
                                                                              Add Wechatupowcoder
```

```
Value visit (LambdaExp e, Env env) {
  return new Value.FunVal(env, e.formals(), e.body());
}
```

Assignment Project Exam Help



Add WeChat powcoder

Evaluate a Call Expression

```
(define identity
(lambda (x) x)
```

SSIGNMENT Project Exam Help

function value. For example, for the call expression (identity i) the variable expression identity's value will be the function value.

2. Evaluate operands. For each expression that is in place of a formal parameter, evaluate it to a value. For example, for the call expression (identity i) the variable expression i's value will be the only opened alte. We Chat powcoder

- 3. Evaluate function body. This step has three parts.
 - a) Find the expression that is the body of the function value,
 - b) create a suitable environment for that body to evaluate, and
 - c) evaluate the body.

Evaluate a Call Expression

Assignment Project Exam Help

Add WeChat powcoder

Dynamic Errors in FuncLang

Errors that cannot be found using grammar rules:

Assignment and parameters and actual parameters do not match the language transport the language transport to the language transport transport transport to the language transport transport transport t

▶ if exp (operator) does not return a function value

Implementation: Evaluating a Call Expression

```
Value visit (CallExp e, Env env) {
                          //Step 1: Evaluate operator
                          Object result = e.operator().accept(this, env);
               niment relived in the Project Exam Help return new Value. Dynamic Erro ("Operator not a function");
                         Value.FunVal operator = (Value.FunVal) result;
                          List \langle Exp \rangle operands = e.operands();
                                 PS: Evaluate on two coder com
                         for (Exp exp : operands)
                                 actuals.add((Value)exp.accept(this, env));
Add in which the control of the cont
                         if (formals.size()!=actuals.size())
                                return new Value.DynamicError("Argument mismatch in call ");
                         Env fenv = appendEnv(operator.env(), initEnv);
                         for (int i = 0; i < formals. size(); i++)
                            fenv = new ExtendEnv(fenv, formals.get(i), actuals.get(i));
                     return (Value) operator.body().accept(this, fenv);
```

Control Structure: Extending Value

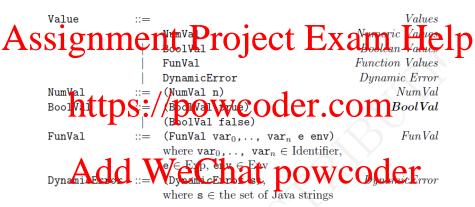


Figure 5.7: The set of Legal Values for the Funciang Language with new boolean value

Control Structure: Semantic Rules

VALUE OF GREATEREXP

value exp₀ env = (NumValata₀)

Value (GreaterExp exp₀ exp₁) env = (BoolVal b)

Value (GreaterExp exp₀ exp₁) env = (BoolVal b)

```
\begin{array}{c} \textbf{https.}_{\text{value}} \text{ of Expaces } \\ \textbf{value} \text{ exp}_1 \text{ env} = (\texttt{NumVal} \ \texttt{n}_1) & \texttt{n}_0 == \texttt{n}_1 = \texttt{b} \\ \hline \textbf{value} \text{ (EqualExp exp} 0 \text{ exp}_1) \text{ ehv} = (\texttt{BoolVal} \ \texttt{b}) \\ \textbf{Add}_{\text{of Lessex}} \textbf{Chat powcoder} \\ \textbf{value} \text{ exp}_0 \text{ env} = (\texttt{NumVal} \ \texttt{n}_0) \\ \hline \textbf{value} \text{ exp}_1 \text{ env} = (\texttt{NumVal} \ \texttt{n}_1) & \texttt{n}_0 < \texttt{n}_1 = \texttt{b} \\ \hline \textbf{value} \text{ (LessExp exp} 0 \text{ exp}_1) \text{ env} = (\texttt{BoolVal} \ \texttt{b}) \\ \hline \end{array}
```

Control Structure: Semantic Rules

Assignment Project Exam Help

Add WeChat powcoder

Semantics of List: Extending the Values

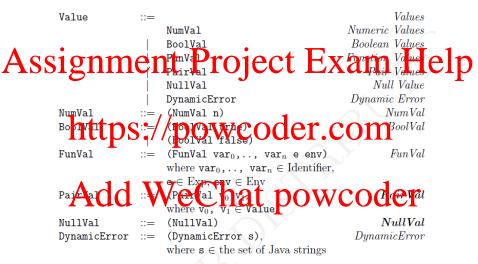


Figure 5.9: The set of Legal Values for the Funclang Language with new pair and null values

Semantics for List Operations

Assignment Project Exam Help

value \exp_0 env = val_0 , ..., $\operatorname{value} \exp_n$ env = val_n

https://powcoder.com

A corollary of the relation is:

Add wa We Chat powe oder

```
Note.
exp_0 ... | |val_1 |
exp_1 ... | |val_2 |
... |
exp_n = |val_n |
```

Semantics for List Operations

The value of a CarExp is given by:

```
value (CarExp exp) env = val
```

Assignment Project Exam Help

value (CdrExp exp) env = lval

http:sex/ on pitywc a where exp Exp env env The value of a nsExp is given by:

value (ConsExp exp exp') env = (ListVal val lval)

Add we Chat powcoder

value exp' env = 1va

The value of a NullExp is given by:

```
value (NullExp exp) env = #t if value exp env = (EmptyList) value \  \, (NullExp \ exp) \ env = #f \\ if value exp env = (ListVal \ val \ lval') \ where \ lval' \in ListVal \\ where \ exp \in Exp \ env \in Env
```



Review

Function: Function ssignment Project Exam Help

- Lambda Expression, Call Expression
- Combine with Let and Define: functions are also variables
- if the trees: Condicipo w coder.com
- Understanding of pair and list: List is a pair, pair is not a list
- syntax: CFG, semantic: operational
- rectand and a transfer of the transfer of the
- ► Values: NumVal, FunVal, PairVal, NullVal, BoolVal, UnitVal, Dynamic Errors