MSDscript

Generated by Doxygen 1.9.6

1 MSDScript 1

2 Hierarchical Index 3

2.1 Class Hierarchy . . . 3

3 Class Index 5

3.1 Class List . . . 5

4 File Index 7

4.1 File List . . . 7

5 Class Documentation 9

5.1 Add Class Reference . . . 9

5.1.1 Detailed Description . . . 10

5.1.2 Constructor & Destructor Documentation . . . 10

5.1.2.1 Add() . . . 10

5.1.3 Member Function Documentation . . . 10

5.1.3.1 equals() . . . 10

5.1.3.2 has_variable() . . . 11

5.1.3.3 interp() . . . 11

5.1.3.4 subst() . . . 11

5.2 Expr Class Reference . . . 12

5.2.1 Detailed Description . . . 12

5.2.2 Member Function Documentation . . . 12

5.2.2.1 equals() . . . 12

5.2.2.2 has_variable() . . . 13

5.2.2.3 interp() . . . 13

5.2.2.4 subst() . . . 13

5.3 Mult Class Reference . . . 14

5.3.1 Detailed Description . . . 15

5.3.2 Constructor & Destructor Documentation . . . 15

5.3.2.1 Mult() . . . 15

5.3.3 Member Function Documentation . . . 15

5.3.3.1 equals() . . . 15

5.3.3.2 has_variable() . . . 16

5.3.3.3 interp() . . . 16

5.3.3.4 subst() . . . 16

5.4 Num Class Reference . . . 17

5.4.1 Detailed Description . . . 18

5.4.2 Constructor & Destructor Documentation . . . 18

5.4.2.1 Num() . . . 18

5.4.3 Member Function Documentation . . . 18

5.4.3.1 equals() . . . 18

5.4.3.2 has_variable() . . . 19

5.4.3.3 interp() 19
5.4.3.4 subst() 19
5.5 Variable Class Reference 20
5.5.1 Detailed Description 20
5.5.2 Constructor & Destructor Documentation 20
5.5.2.1 Variable() 20
5.5.3 Member Function Documentation 21
5.5.3.1 equals() 21
5.5.3.2 has_variable() 21
5.5.3.3 interp() 21
5.5.3.4 subst() 22
6 File Documentation **23**
6.1 /Users/rasonhung/Study/MSD/CS6015/MSDScript/cmdline.h File Reference 23
6.1.1 Detailed Description 23
6.1.2 Function Documentation 23
6.1.2.1 use_arguments() 23
6.2 /Users/rasonhung/Study/MSD/CS6015/MSDScript/cmdline.h 24
6.3 /Users/rasonhung/Study/MSD/CS6015/MSDScript/expr.h File Reference 24
6.3.1 Detailed Description 25
6.4 /Users/rasonhung/Study/MSD/CS6015/MSDScript/expr.h 25
Index **27**

Chapter 1

MSDScript

passing arguments through command line, execute with –help, –test ...

Author

Juisheng Hung (Rason)

Date

01-16-2023

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Expr 12
    Add 9
    Mult 14
    Num 17
    Variable 20

Chapter 3

Class Index

3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Add

Add class inherits from Expr class, representing addition for two expressions 9

Expr

Abstract expression class
(pure abstract class) 12

Mult

Mult class inherits from Expr class, representing multiplication for two expressions 14

Num

Num class inherits from Expr class, representing pure number 17

Variable

Variable class inherits from Expr class, representing pure variable 20

Chapter 4

File Index

4.1 File List

Here is a list of all documented files with brief descriptions:

/Users/rasonhung/Study/MSD/CS6015/MSDScript/cmdline.h
Actual function that executes command line script 23
/Users/rasonhung/Study/MSD/CS6015/MSDScript/expr.h
Expression class 24

Chapter 5

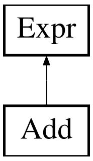
Class Documentation

5.1 Add Class Reference

Add class inherits from Expr class, representing addition for two expressions.

`#include <expr.h>`

Inheritance diagram for Add:



Public Member Functions

- Add (Expr ∗lhs, Expr ∗rhs)

  Constructor for Add object.
- bool equals (Expr ∗e) override

  Judge if this Add class object equals to another object.
- int interp () override

  Interpret Add object to an integer value.
- bool has_variable () override

  Judge if the Add object contains any Variable.
- Expr ∗ subst (std::string string, Expr ∗e) override

  Substitute the Variable inside Add object with another Expr.

- virtual bool equals (Expr ∗e)=0

  Judge if this Expr class object equals to another object.
- virtual int interp ()=0

  Interpret Expr object to an integer value.
- virtual bool has_variable ()=0

  Judge if the Expr object contains any Variable.
- virtual Expr ∗ subst (std::string string, Expr ∗e)=0

  Substitute the Variable inside Expr object with another Expr.

Public Attributes

- Expr ∗ **lhs**

  the Expr object that makes up the left hand side of the Add object
- Expr ∗ **rhs**

  the Expr object that makes up the right hand side of the Add object

5.1.1 Detailed Description

Add class inherits from Expr class, representing addition for two expressions.

5.1.2 Constructor & Destructor Documentation

5.1.2.1 Add()

```
Add::Add (
            Expr * lhs,
            Expr * rhs )
```

Constructor for Add object.

Parameters

lhs	an Expr object on the left hand side
rhs	an Expr object on the right hand side

5.1.3 Member Function Documentation

5.1.3.1 equals()

```
bool Add::equals (
            Expr * e )  [override], [virtual]
```

Judge if this Add class object equals to another object.

Parameters

e	an Expr pointer to Expr object waited to be compared

Returns

returns a boolean, true if two object equals, otherwise false

Implements Expr.

5.1.3.2 has_variable()

```
bool Add::has_variable ( )  [override], [virtual]
```

Judge if the Add object contains any Variable.

Returns

returns a boolean, true if the Expr object contains any Variable, otherwise false

Implements Expr.

5.1.3.3 interp()

```
int Add::interp ( )  [override], [virtual]
```

Interpret Add object to an integer value.

Returns

returns the actual integer value (lhs + rhs) of the Add, if it contains Variable, throw an exception

Implements Expr.

5.1.3.4 subst()

```
Expr * Add::subst (
             std::string string,
             Expr * e )  [override], [virtual]
```

Substitute the Variable inside Add object with another Expr.

Parameters

string	first argument, a target string that is waited to be substituted
e	second argument, an Expr pointer to object that is going to substitute the Variable inside expression

Returns

returns the new Expr pointer to object after substitution, return the original object if string Variable not found

Implements Expr.

The documentation for this class was generated from the following files:

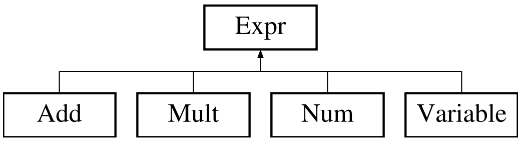
- /Users/rasonhung/Study/MSD/CS6015/MSDScript/expr.h
- /Users/rasonhung/Study/MSD/CS6015/MSDScript/expr.cpp

5.2 Expr Class Reference

Abstract expression class
(pure abstract class)

```
#include <expr.h>
```

Inheritance diagram for Expr:



Public Member Functions

- virtual bool equals (Expr ∗e)=0

  Judge if this Expr class object equals to another object.
- virtual int interp ()=0

  Interpret Expr object to an integer value.
- virtual bool has_variable ()=0

  Judge if the Expr object contains any Variable.
- virtual Expr ∗ subst (std::string string, Expr ∗e)=0

  Substitute the Variable inside Expr object with another Expr.

5.2.1 Detailed Description

Abstract expression class
(pure abstract class)

5.2.2 Member Function Documentation

5.2.2.1 equals()

```
virtual bool Expr::equals (
            Expr ∗ e )  [pure virtual]
```

Judge if this Expr class object equals to another object.

Parameters

e	an Expr pointer to object waited to be compared

Returns

returns a boolean, true if two object equals, otherwise false

Implemented in Num, Add, Mult, and Variable.

5.2.2.2 has_variable()

```
virtual bool Expr::has_variable ( )  [pure virtual]
```

Judge if the Expr object contains any Variable.

Returns

returns a boolean, true if the Expr object contains any Variable, otherwise false

Implemented in Num, Add, Mult, and Variable.

5.2.2.3 interp()

```
virtual int Expr::interp ( )  [pure virtual]
```

Interpret Expr object to an integer value.

Returns

returns the actual integer value of the Expr, if it contains Variable, throw an exception

Implemented in Num, Add, Mult, and Variable.

5.2.2.4 subst()

```
virtual Expr * Expr::subst (
             std::string string,
             Expr * e )  [pure virtual]
```

Substitute the Variable inside Expr object with another Expr.

Parameters

string	first argument, a target string that is waited to be substituted
e	second argument, an Expr pointer to object that is going to substitute the Variable inside expression

Returns

returns the new Expr pointer to object after substitution, return the original object if string Variable not found

Implemented in Num, Add, Mult, and Variable.

The documentation for this class was generated from the following file:

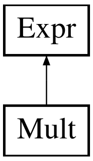
- /Users/rasonhung/Study/MSD/CS6015/MSDScript/expr.h

5.3 Mult Class Reference

Mult class inherits from Expr class, representing multiplication for two expressions.

```
#include <expr.h>
```

Inheritance diagram for Mult:



Public Member Functions

- Mult (Expr ∗lhs, Expr ∗rhs)

  Constructor for Mult object.
- bool equals (Expr ∗e) override

  Judge if this Mult class object equals to another object.
- int interp () override

  Interpret Mult object to an integer value.
- bool has_variable () override

  Judge if the Mult object contains any Variable.
- Expr ∗ subst (std::string string, Expr ∗e) override

  Substitute the Variable inside Mult object with another Expr.

- virtual bool equals (Expr ∗e)=0

  Judge if this Expr class object equals to another object.
- virtual int interp ()=0

  Interpret Expr object to an integer value.
- virtual bool has_variable ()=0

  Judge if the Expr object contains any Variable.
- virtual Expr ∗ subst (std::string string, Expr ∗e)=0

  Substitute the Variable inside Expr object with another Expr.

Public Attributes

- Expr ∗ **lhs**

  the Expr object that makes up the left hand side of the Mult object
- Expr ∗ **rhs**

  the Expr object that makes up the right hand side of the Mult object

5.3.1 Detailed Description

Mult class inherits from Expr class, representing multiplication for two expressions.

5.3.2 Constructor & Destructor Documentation

5.3.2.1 Mult()

```
Mult::Mult (
            Expr * lhs,
            Expr * rhs )
```

Constructor for Mult object.

Parameters

lhs	an Expr object on the left hand side
rhs	an Expr object on the right hand side

5.3.3 Member Function Documentation

5.3.3.1 equals()

```
bool Mult::equals (
            Expr * e )  [override], [virtual]
```

Judge if this Mult class object equals to another object.

Parameters

e	an Expr pointer to Expr object waited to be compared

Returns

returns a boolean, true if two object equals, otherwise false

Implements Expr.

5.3.3.2 has_variable()

```
bool Mult::has_variable ( )  [override], [virtual]
```

Judge if the Mult object contains any Variable.

Returns

returns a boolean, true if the Expr object contains any Variable, otherwise false

Implements Expr.

5.3.3.3 interp()

```
int Mult::interp ( )  [override], [virtual]
```

Interpret Mult object to an integer value.

Returns

returns the actual integer value (lhs ∗ rhs) of the Mult, if it contains Variable, throw an exception

Implements Expr.

5.3.3.4 subst()

```
Expr * Mult::subst (
             std::string string,
             Expr * e )  [override], [virtual]
```

Substitute the Variable inside Mult object with another Expr.

Parameters

string	first argument, a target string that is waited to be substituted
e	second argument, an Expr pointer to object that is going to substitute the Variable inside expression

Returns

returns the new Expr pointer to object after substitution, return the original object if string Variable not found

Implements Expr.

The documentation for this class was generated from the following files:

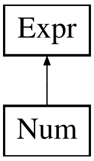
- /Users/rasonhung/Study/MSD/CS6015/MSDScript/expr.h
- /Users/rasonhung/Study/MSD/CS6015/MSDScript/expr.cpp

5.4 Num Class Reference

Num class inherits from Expr class, representing pure number.

```
#include <expr.h>
```

Inheritance diagram for Num:



Public Member Functions

- Num (int val)

  Constructor for Num object.
- bool equals (Expr ∗e) override

  Judge if this Num class object equals to another object.
- int interp () override

  Interpret Num object to an integer value.
- bool has_variable () override

  Judge if the Num object contains any Variable.
- Expr ∗ subst (std::string string, Expr ∗e) override

  Substitute the Variable inside Num object with another Expr.

- virtual bool equals (Expr ∗e)=0

  Judge if this Expr class object equals to another object.
- virtual int interp ()=0

  Interpret Expr object to an integer value.
- virtual bool has_variable ()=0

  Judge if the Expr object contains any Variable.
- virtual Expr ∗ subst (std::string string, Expr ∗e)=0

  Substitute the Variable inside Expr object with another Expr.

Public Attributes

- int **val**

  the integer value of the Num object

5.4.1 Detailed Description

Num class inherits from Expr class, representing pure number.

5.4.2 Constructor & Destructor Documentation

5.4.2.1 Num()

```
Num::Num (
            int val )  [explicit]
```

Constructor for Num object.

Parameters

val	integer value of Num

5.4.3 Member Function Documentation

5.4.3.1 equals()

```
bool Num::equals (
            Expr * e )  [override], [virtual]
```

Judge if this Num class object equals to another object.

Parameters

e	an Expr pointer to Expr object waited to be compared

Returns

returns a boolean, true if two object equals, otherwise false

Implements Expr.

5.4.3.2 has_variable()

```
bool Num::has_variable ( )  [override], [virtual]
```

Judge if the Num object contains any Variable.

Returns

returns a boolean, always return false

Implements Expr.

5.4.3.3 interp()

```
int Num::interp ( )  [override], [virtual]
```

Interpret Num object to an integer value.

Returns

returns the actual integer value of the Num

Implements Expr.

5.4.3.4 subst()

```
Expr * Num::subst (
             std::string string,
             Expr * e )  [override], [virtual]
```

Substitute the Variable inside Num object with another Expr.

Parameters

string	first argument, a target string that is waited to be substituted
e	second argument, an Expr pointer to object that is going to substitute the Variable inside expression

Returns

returns this object, since there is no Variable in Num object

Implements Expr.

The documentation for this class was generated from the following files:

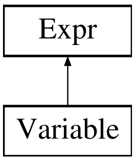
- /Users/rasonhung/Study/MSD/CS6015/MSDScript/expr.h
- /Users/rasonhung/Study/MSD/CS6015/MSDScript/expr.cpp

5.5 Variable Class Reference

Variable class inherits from Expr class, representing pure variable.

```
#include <expr.h>
```

Inheritance diagram for Variable:



Public Member Functions

- Variable (std::string varName)

  Constructor for Variable object.
- bool equals (Expr ∗e) override

  Judge if this Variable class object equals to another object, overrides function in superclass.
- int interp () override

  Interpret Variable object to an integer value.
- bool has_variable () override

  Judge if the Variable object contains any Variable.
- Expr ∗ subst (std::string string, Expr ∗e) override

  Substitute the Variable object with another Expr.

- virtual bool equals (Expr ∗e)=0

  Judge if this Expr class object equals to another object.
- virtual int interp ()=0

  Interpret Expr object to an integer value.
- virtual bool has_variable ()=0

  Judge if the Expr object contains any Variable.
- virtual Expr ∗ subst (std::string string, Expr ∗e)=0

  Substitute the Variable inside Expr object with another Expr.

Public Attributes

- std::string **name**

  the string name that makes up the Variable object

5.5.1 Detailed Description

Variable class inherits from Expr class, representing pure variable.

5.5.2 Constructor & Destructor Documentation

5.5.2.1 Variable()

```
Variable::Variable (
            std::string varName )  [explicit]
```

Constructor for Variable object.

Parameters

varName	a string that can be seen as the label of the Variable

5.5.3 Member Function Documentation

5.5.3.1 equals()

```
bool Variable::equals (
            Expr * e )  [override], [virtual]
```

Judge if this Variable class object equals to another object, overrides function in superclass.

Parameters

e	an Expr pointer to Expr object waited to be compared

Returns

returns a boolean, true if two object equals, otherwise false

Implements Expr.

5.5.3.2 has_variable()

```
bool Variable::has_variable ( )  [override], [virtual]
```

Judge if the Variable object contains any Variable.

Returns

returns a boolean, always return true

Implements Expr.

5.5.3.3 interp()

```
int Variable::interp ( )  [override], [virtual]
```

Interpret Variable object to an integer value.

Returns

A Variable doesn't have specific integer value, throw an exception

Implements Expr.

5.5.3.4 subst()

```
Expr * Variable::subst (
            std::string string,
            Expr * e )  [override], [virtual]
```

Substitute the Variable object with another Expr.

Parameters

string	first argument, a target string that is waited to be substituted
e	second argument, an Expr pointer to object that is going to substitute the Variable inside expression

Returns

returns the new Expr pointer to object after substitution, return the original object if string Variable not found

Implements Expr.

The documentation for this class was generated from the following files:

- /Users/rasonhung/Study/MSD/CS6015/MSDScript/expr.h
- /Users/rasonhung/Study/MSD/CS6015/MSDScript/expr.cpp

Chapter 6

File Documentation

6.1 /Users/rasonhung/Study/MSD/CS6015/MSDScript/cmdline.h File Reference

actual function that executes command line script

```
#include <iostream>
#include <string>
```

Functions

- void use_arguments (int argc, const char ∗argv[ ])

  Take arguments from command line as input, execute corresponding output as required.

6.1.1 Detailed Description

actual function that executes command line script

6.1.2 Function Documentation

6.1.2.1 use_arguments()

```
void use_arguments (
             int argc,
             const char * argv[ ] )
```

Take arguments from command line as input, execute corresponding output as required.

Parameters

argc	first argument, the integer number of arguments passed into
argv	second argument, the pointer to the array of characters that is passed into as parameter

Returns

returns void
"--help": if it is the next argument after program name, print out help message, and do not examine other arguments
"--test": if it is the only argument after program name, then print out test result, otherwise, will be treated as invalid argument input any other strings as input: invalid argument, exit the program with 1

6.2 /Users/rasonhung/Study/MSD/CS6015/MSDScript/cmdline.h

Go to the documentation of this file.

```
//
//  cmdline.h
//  CommandLine
//
//  Created by Rason Hung on 1/16/23.
//

#pragma include once
#include <iostream>
#include <string>

void use_arguments(int argc, const char *argv[]);
```

6.3 /Users/rasonhung/Study/MSD/CS6015/MSDScript/expr.h File Reference

expression class

```
#include <cstdio>
#include <string>
```

Classes

- class Expr

  Abstract expression class
  (pure abstract class)
- class Num

  Num class inherits from Expr class, representing pure number.
- class Add

  Add class inherits from Expr class, representing addition for two expressions.
- class Mult

  Mult class inherits from Expr class, representing multiplication for two expressions.
- class Variable

  Variable class inherits from Expr class, representing pure variable.

6.3.1 Detailed Description

expression class

Contains the blueprint of the superclass - Expr, with its subclass - Num, Add, Mult, Variable

6.4 /Users/rasonhung/Study/MSD/CS6015/MSDScript/expr.h

Go to the documentation of this file.

```
//
//  expr.h
//  ExpressionClasses
//
//  Created by Rason Hung on 1/22/23.
//

#pragma include once
#include <cstdio>
#include <string>


class Expr {
public:
    virtual bool equals(Expr *e) = 0;

    virtual int interp() = 0;

    virtual bool has_variable() = 0;

    virtual Expr* subst(std::string string, Expr* e)=0;
};

class Num : public Expr {
public:
    int val;

    explicit Num(int val);

    bool equals(Expr *e) override;

    int interp() override;

    bool has_variable() override;

    Expr* subst(std::string string, Expr* e) override;
};


class Add : public Expr {
public:
    Expr *lhs;
    Expr *rhs;

    Add(Expr *lhs, Expr *rhs);

    bool equals(Expr *e) override;

    int interp() override;

    bool has_variable() override;

    Expr* subst(std::string string, Expr* e) override;
};

class Mult : public Expr {
public:
    Expr *lhs;
    Expr *rhs;

    Mult(Expr *lhs, Expr *rhs);

    bool equals(Expr *e) override;

    int interp() override;

    bool has_variable() override;

    Expr* subst(std::string string, Expr* e) override;
};

class Variable : public Expr {
public:
    std::string name;

    explicit Variable(std::string varName);

    bool equals(Expr *e) override;

    int interp() override;

    bool has_variable() override;

    Expr* subst(std::string string, Expr* e) override;
};
```

Index

/Users/rasonhung/Study/MSD/CS6015/MSDScript/cmdline.h, 23
/Users/rasonhung/Study/MSD/CS6015/MSDScript/expr.h, 24

Add, 9
- Add, 10
- equals, 10
- has_variable, 11
- interp, 11
- subst, 11

cmdline.h
- use_arguments, 23

equals
- Add, 10
- Expr, 12
- Mult, 15
- Num, 18
- Variable, 21

Expr, 12
- equals, 12
- has_variable, 13
- interp, 13
- subst, 13

has_variable
- Add, 11
- Expr, 13
- Mult, 16
- Num, 18
- Variable, 21

interp
- Add, 11
- Expr, 13
- Mult, 16
- Num, 19
- Variable, 21

Mult, 14
- equals, 15
- has_variable, 16
- interp, 16
- Mult, 15
- subst, 16

Num, 17
- equals, 18
- has_variable, 18
- interp, 19
- Num, 18
- subst, 19

subst
- Add, 11
- Expr, 13
- Mult, 16
- Num, 19
- Variable, 21

use_arguments
- cmdline.h, 23

Variable, 20
- equals, 21
- has_variable, 21
- interp, 21
- subst, 21
- Variable, 20