CSE110A: Compilers

March 30, 2022

- **Topics**:
  - *Logistics*
    - Location
    - Office hours
    - Piazza
    - Quiz
  - *Compiler Overview*
    - What is a compiler
    - What are the different stages of a compiler
      - Frontend
      - Intermediate
      - Backend

Announcements

- New room!
  - Previous room only sat 64 students
  - Moving to Soc Sci 2 - 71 (Bigger room)
    - should help with social distancing as well
  - We let everyone on the wait list in (68 students)
    - Class is slightly more impacted
    - might effect grading time, office hours, etc. but we will do our best

Announcements

- Piazza is up and going!
- Private questions for homework help (especially if you are going to share code, or ask for clarification on a grade)
- Public questions for framework, programming languages, content, etc.
- Did anyone set up a discord?

Announcements

Office hours:

- Yanwen:
  - ??
- Arrian:
  - ??
- Neal:
  - ??

Quiz

Background

What classes have you taken:

CSE 103	18 respondents	**50 %**	
CSE 120	32 respondents	89 %	
CSE 130	27 respondents	75 %	
No Answer	1 respondent	3 %	

Background

Have you programmed in Python before?

It is worthwhile to learn!

https://www.tiobe.com/tiobe-index/

What do people hope to get out of this class?

A few answers that I liked:

- “I don’t know too much about compilers and I want to learn!”
- ”learning about compilers will make me a better programmer”
- ”Increase knowledge about computer science”
- “Want to make my own programming language”
- “Why programming languages are the way they are”

Quiz

- Thank you for all your thoughtful answers!

- We will decide on what to do about masks later

Review

- Normally we would do a review here, but nothing too much to review

Schedule

- Introduction to compilers
- Compiler architecture

Schedule

- **Introduction to compilers**
- Compiler architecture

What is a compiler?

Let’s discuss

What is a compiler?

Let’s discuss

What are some of your favorite compilers

Let’s discuss

```
---
title: "Fundamentals of Compiler Design"
layout: single
---


### Welcome to **CSE110A:** _Fundamentals of Compiler Design_, Spring 2022 Quarter at UCSC!

- **Instructor:** [Tyler Sorensen](https://users.soe.ucsc.edu/~tsorensen/)
- **Time:** Mondays, Wednesdays and Fridays: 4:00 - 5:05 pm
- **Location:** Porter 144

Hello and welcome to the fundamentals of compiler design class!

In this class you will learn about compiler design and implementation. In the abstract, compilers explore many of the [foundational problems in computer science](https://en.wikipedia.org/wiki/Halting_problem). In practice, compilers are [massive pieces of well-oiled software](https://www.phoronix.com/scan.php?page=news_item&px=MTg3OTQ), and are some of the engineering marvels of the modern world.

_COVID Note_ : The last few years have been difficult due to the COVID pandemic. Public health concerns and policies remain volatile. The first priority in this class in your health and well-being. We will approach any challenges that arise with compassion and understanding. I expect that you will do the same, both to the teaching staff and to your classmates. We will follow university guidelines and work together to have a productive and fun quarter.

```

Fundamentals of Compiler Design

Welcome to CSE110A: *Fundamentals of Compiler Design*, Spring 2022 Quarter at UCSC!

- **Instructor:** Tyler Sorensen
- **Time:** Mondays, Wednesdays and Fridays: 4:00 - 5:05 pm
- **Location:** Porter 144

Hello and welcome to the fundamentals of compiler design class!

In this class you will learn about compiler design and implementation. In the abstract, compilers explore many of the foundational problems in computer science. In practice, compilers are massive pieces of well-oiled software, and are some of the engineering marvels of the modern world.

Building this website started with:

- Markdown to describe the page
- compiled with Jekyll to a static webpage
- static webpage is in HTML and javascript


Canon

What is a compiler?

What is a compiler?

This is way too general to be useful
Any program fits this description.

What is a compiler?

A theoretical answer

```
---
title: "Fundamentals of Compiler Design"
layout: single
---


### Welcome to **CSE110A:** _Fundamentals of Compiler Design_, Spring 2022 Quarter at UCSC!

- **Instructor:** [Tyler Sorensen](https://users.soe.ucsc.edu/~tsorensen/)
- **Time:** Mondays, Wednesdays and Fridays: 4:00 - 5:05 pm
- **Location:** Porter 144

Hello and welcome to the fundamentals of compiler design class!

In this class you will learn about compiler design and implementation. In the abstract, compilers explore many of the [foundational problems in computer science](https://en.wikipedia.org/wiki/Halting_problem). In practice, compilers are [massive pieces of well-oiled software](https://www.phoronix.com/scan.php?page=news_item&px=MTg3OTQ), and are some of the engineering marvels of the modern world.

_COVID Note_ : The last few years have been difficult due to the COVID pandemic. Public health concerns and policies remain volatile. The first priority in this class in your health and well-being. We will approach any challenges that arise with compassion and understanding. I expect that you will do the same, both to the teaching staff and to your classmates. We will follow university guidelines and work together to have a productive and fun quarter.
```

Fundamentals of Compiler Design

Welcome to CSE110A: *Fundamentals of Compiler Design*, Spring 2022 Quarter at UCSC!

- **Instructor:** Tyler Sorensen
- **Time:** Mondays, Wednesdays and Fridays: 4:00 - 5:05 pm
- **Location:** Porter 144

Hello and welcome to the fundamentals of compiler design class!

In this class you will learn about compiler design and implementation. In the abstract, compilers explore many of the foundational problems in computer science. In practice, compilers are massive pieces of well-oiled software, and are some of the engineering marvels of the modern world.

Building this website started with:

- Markdown to describe the page
- compiled with Jekyll to a static webpage
- static webpage is in HTML and javascript

This would be a compiler

What is a compiler?

A more traditional description
What are some examples here?

Strings belonging to language L

A series of statements in programming language L

Strings belonging to language L’

An executable binary file in an ISA language

What is a compiler?

A classic example

Strings belonging to language L		Strings belonging to language L’
A series of statements in programming language L		An executable binary file in an ISA language
A program written in C	gcc/clang	An x86 Binary executable

What is a compiler?

```c
int main() {
  printf("hello world\n");
}
```

`gcc main.c`

Strings belonging to language L

A series of statements in programming language L

A program written in C

gcc/clang

Strings belonging to language L’

An executable binary file in an ISA language

An x86 Binary executable

What is a compiler?

What is wrong with this picture?

```
int main() {
  printf("hello world\n");
}
```

```
$ ./a.out
hello CSE 110A
```

`gcc main.c`

Strings belonging to language L

A series of statements in programming language L

A program written in C

gcc/clang

Strings belonging to language L’

An executable binary file in an ISA language

An x86 Binary executable

What is a compiler?

A valid input must have a equivalent valid output.
Semantic equivalence

Strings belonging to language L

A series of statements in programming language L

A program written in C

gcc/clang

Strings belonging to language L’

An executable binary file in an ISA language

An x86 Binary executable

What is a compiler?

What is wrong with this picture?

```c
int main() {
  printf("hello world\n");
}
```

```
$ ./a.out
hello CSE 110A
```

`gcc main.c`

Strings belonging to language L

A series of statements in programming language L

A program written in C

gcc/clang

Strings belonging to language L’

An executable binary file in an ISA language

An x86 Binary executable

What is a compiler?

```c
int main() {
  printf("hello world\n");
}
```

```
$ ./a.out
hello world
```

`gcc main.c`

Strings belonging to language L

A series of statements in programming language L

A program written in C

gcc/clang

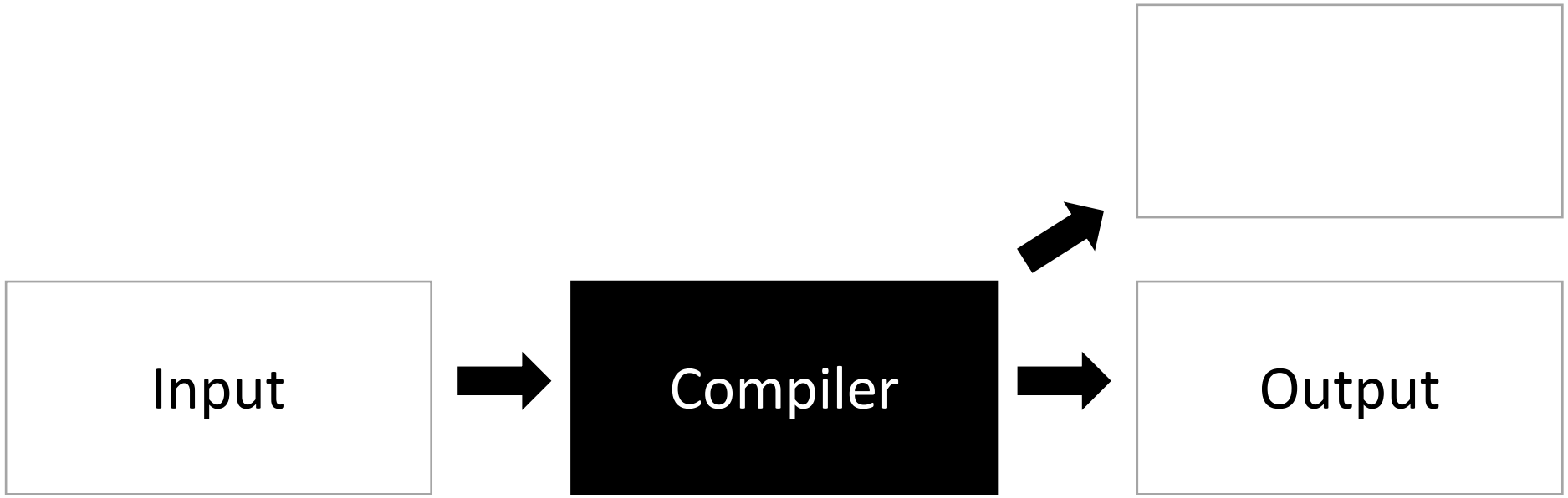
Strings belonging to language L’

An executable binary file in an ISA language

An x86 Binary executable

What is a compiler?

What else does a compiler give you?



Strings belonging to language L

A series of statements in programming language L

A program written in C

gcc/clang

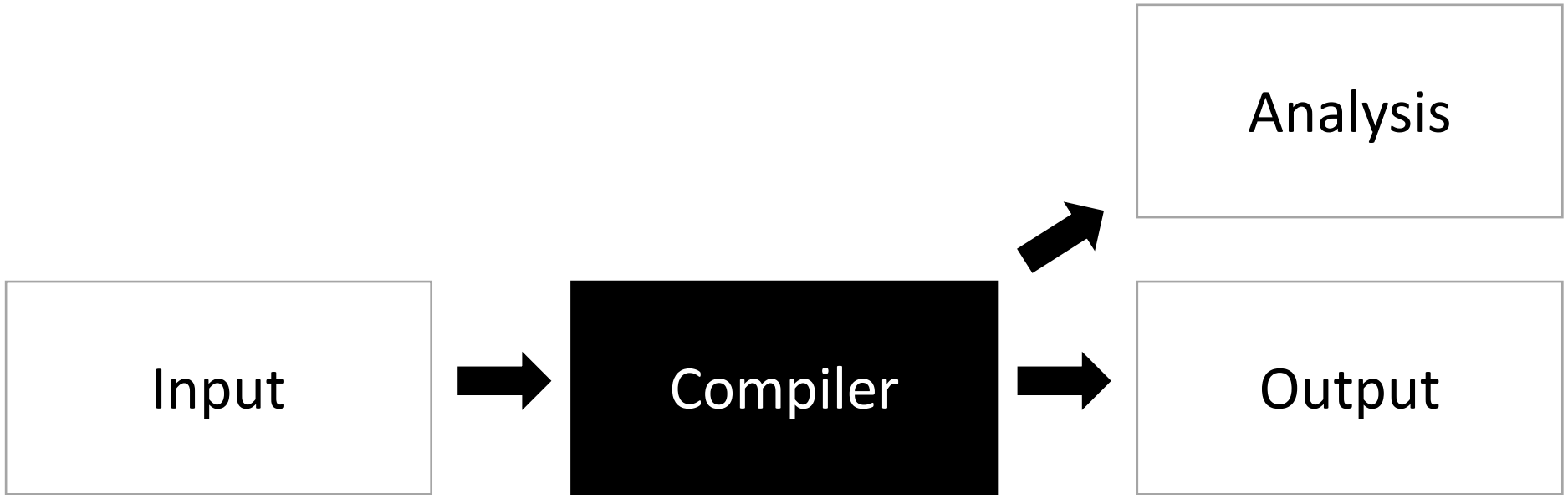
Strings belonging to language L’

An executable binary file in an ISA language

An x86 Binary executable

What is a compiler?

What are some examples here?



Strings belonging to language L

A series of statements in programming language L

A program written in C

gcc/clang

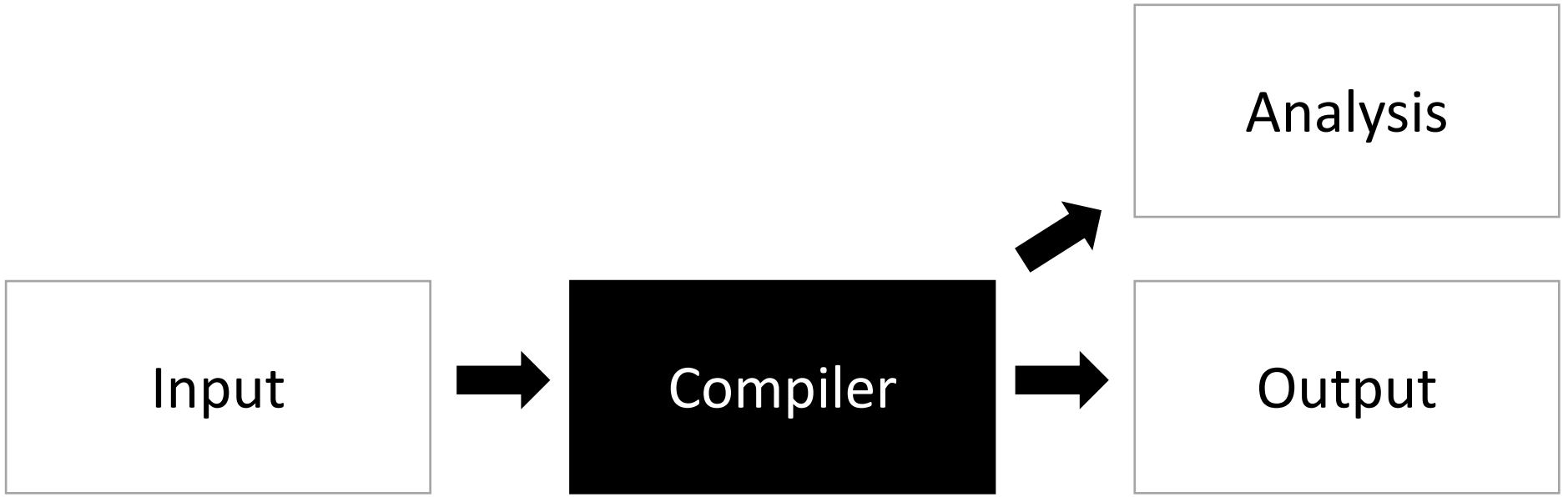
Strings belonging to language L’

An executable binary file in an ISA language

An x86 Binary executable

What is a compiler?

What are some examples here?



Warnings
Errors
Performance logs

Strings belonging to language L

A series of statements in programming language L

A program written in C

gcc/clang

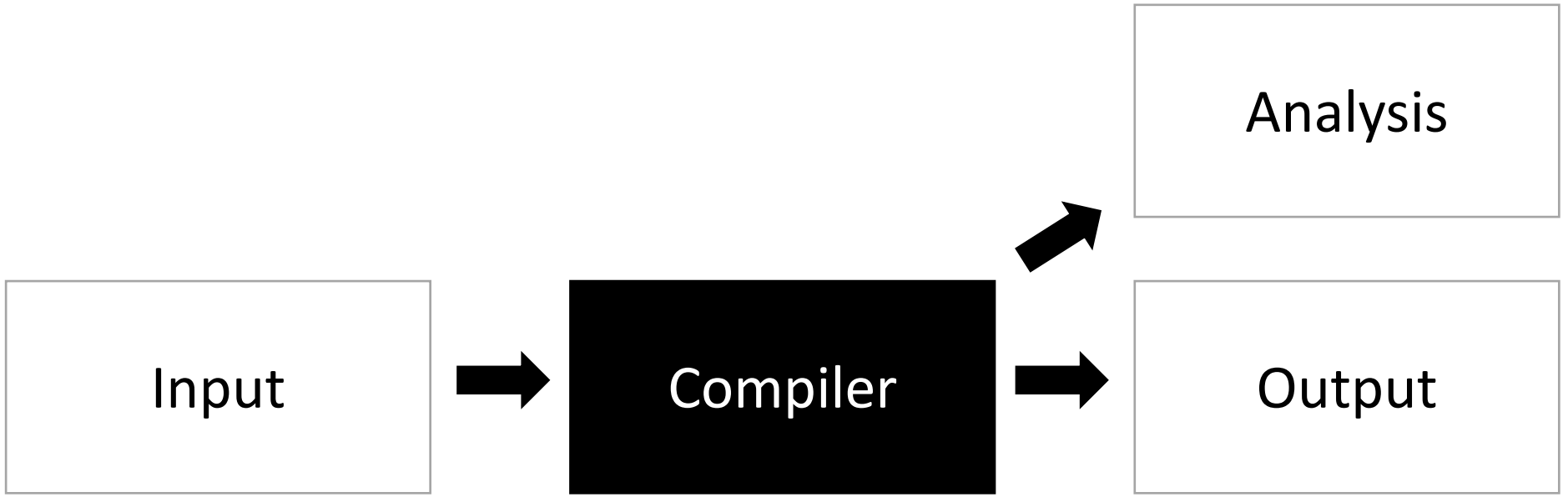
Strings belonging to language L’

An executable binary file in an ISA language

An x86 Binary executable

What is a compiler?

What are some examples here?



Warnings
Errors
Performance logs

Strings belonging to language L

A series of statements in programming language L

A program written in C

gcc/clang

Strings belonging to language L’

An executable binary file in an ISA language

An x86 Binary executable

Demo

- What are some examples of code that might give a warning?

What can happen when the Input isn’t valid?

```
int foo() {
  int x;
  int y = x;
  return y;
}
```

Try running this through the compiler

What can happen when the Input isn't valid?

```
int foo() {
  int x;
  int y = x;
  return y;
}
```

```
int foo(int condition) {
  int x;
  if (condition) {
    x = 5;
  }
  int y = x;
  return y;
}
```

What about this one?

Try running this through the compiler

What is a compiler?

A valid input must have a equivalent valid output.
Semantic equivalence

Strings belonging to language L

A series of statements in programming language L

A program written in C

gcc/clang

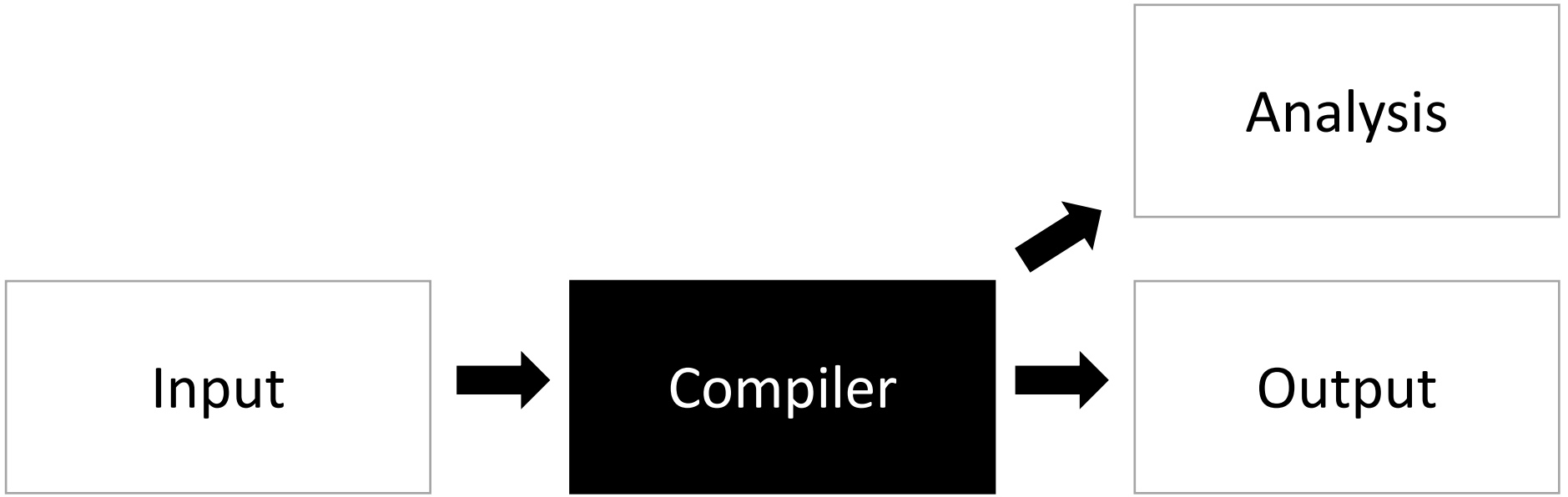
Strings belonging to language L’

An executable binary file in an ISA language

An x86 Binary executable

What is a compiler?

What are some examples here?



Warnings
Errors
Performance logs

Strings belonging to language L

A series of statements in programming language L

A program written in C

gcc/clang

Strings belonging to language L’

An executable binary file in an ISA language

An x86 Binary executable

What can happen when the Input isn’t valid?

```
int foo() {
  int my_var = 5;
  my_var = my_car + 5;
  return my_var
}
```

Try running this through a compiler

What can happen when the Input isn’t valid?

```
int foo() {
  int my_var = 5;
  my_var = my_car + 5;
  return my_var
}
```

Try running this through a compiler

You get an error and a suggestion these days

What can happen when the Input isn't valid?

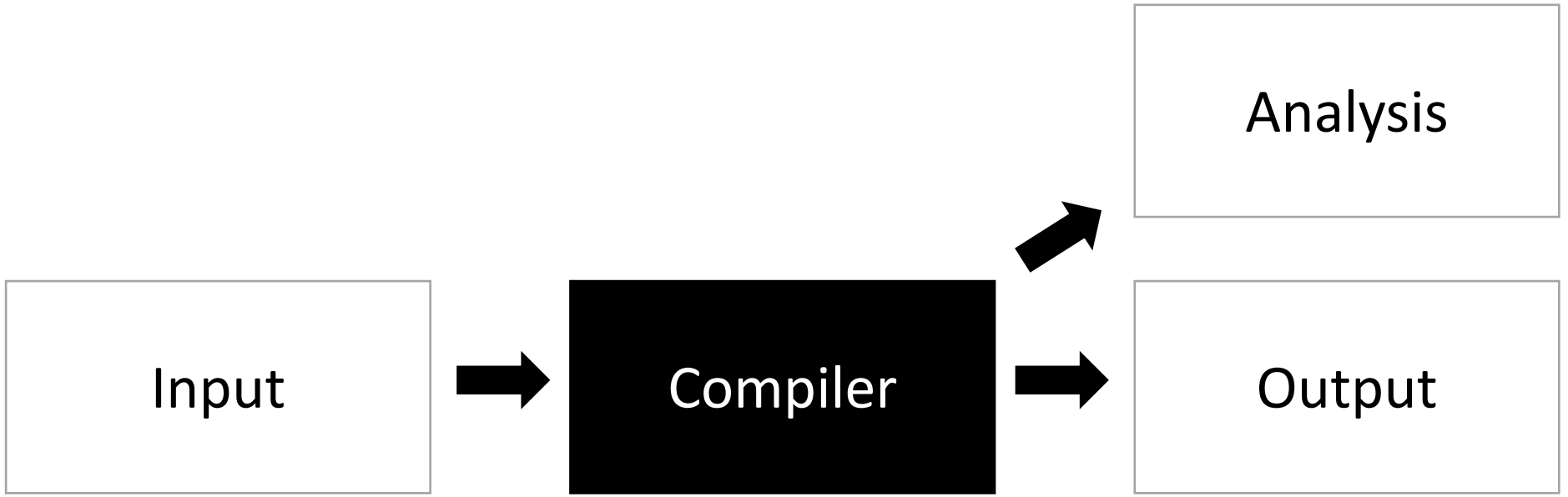
```
int foo() {
  int *x = malloc(100*sizeof(int));
  return x[100];
}
```

What about this one? No error…

What sort of errors are compilers good at catching?
What ones are they not?

What is a compiler?

What are some examples here?



Warnings
Errors
Performance logs

Strings belonging to language L

A series of statements in programming language L

A program written in C

gcc/clang

Strings belonging to language L’

An executable binary file in an ISA language

An x86 Binary executable

How can we know what the compiler is doing?

```
#define SIZE (1024*1024)
int add(int * a, int * b, int * c) {
  for (int i = 0; i < SIZE; i++) {
    a[i] = b[i] + c[i];
  }
  return 0;
}
```

Use the compiler flags

`-Rpass-missed=loop-vectorize`

`-Rpass=loop-vectorize`

Does the compiler need to perform every step?

```
int foo() {
  int my_var = 0;
  for (int i = 0; i < 128; i++) {
    my_var++;
  }
  return my_var;
}
```

Does the compiler need to perform every step?

```
int foo() {
  int my_var = 0;
  for (int i = 0; i < 128; i++) {
    my_var++;
  }
  return my_var;
}
```

Mentally we probably step through the for loop:

Does the compiler need to perform every step?

```
int foo() {
  int my_var = 0;
  for (int i = 0; i < 128; i++) {
    my_var++;
  }
  return my_var;
}
```

Mentally we probably step through the for loop:

What does the compiler do?

What is a compiler?

A valid input must have a equivalent valid output.
Semantic equivalence

Strings belonging to language L

A series of statements in programming language L

A program written in C

gcc/clang

Strings belonging to language L’

An executable binary file in an ISA language

An x86 Binary executable

Does the compiler need to perform every step?

```
int foo() {
  int my_var = 0;
  for (int i = 0; i < 128; i++) {
    my_var++;
  }
  return my_var;
}
```

are these the same?

```
int foo() {
  return 128;
}
```

Does the compiler need to perform every step?

```
int foo() {
  int my_var = 0;
  for (int i = 0; i < 128; i++) {
    my_var++;
  }
  return my_var;
}
```

```
int foo() {
  return 128;
}
```

are these the same?

Functionally - they are the same
Non-functionally - they are not

Most compilers are concerned only with functional equivalence

Schedule

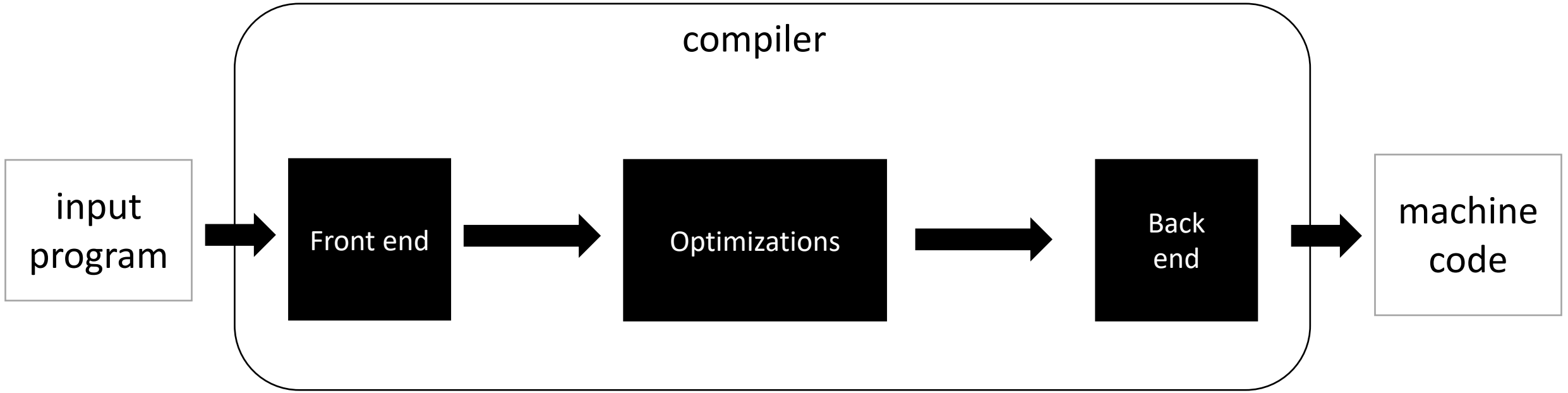
- Introduction to compilers
- **Compiler architecture**

Compiler Architecture

Compiler Architecture

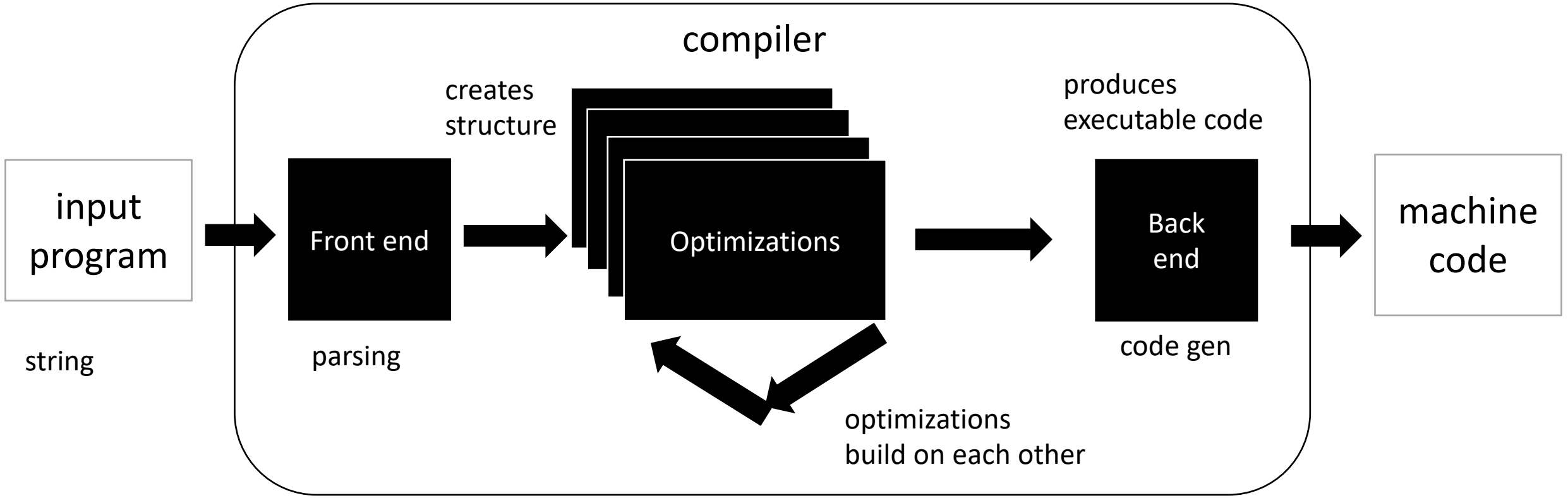
Compilers are complicated and this image is too simple

Compiler Architecture



Medium detailed view

Compiler Architecture



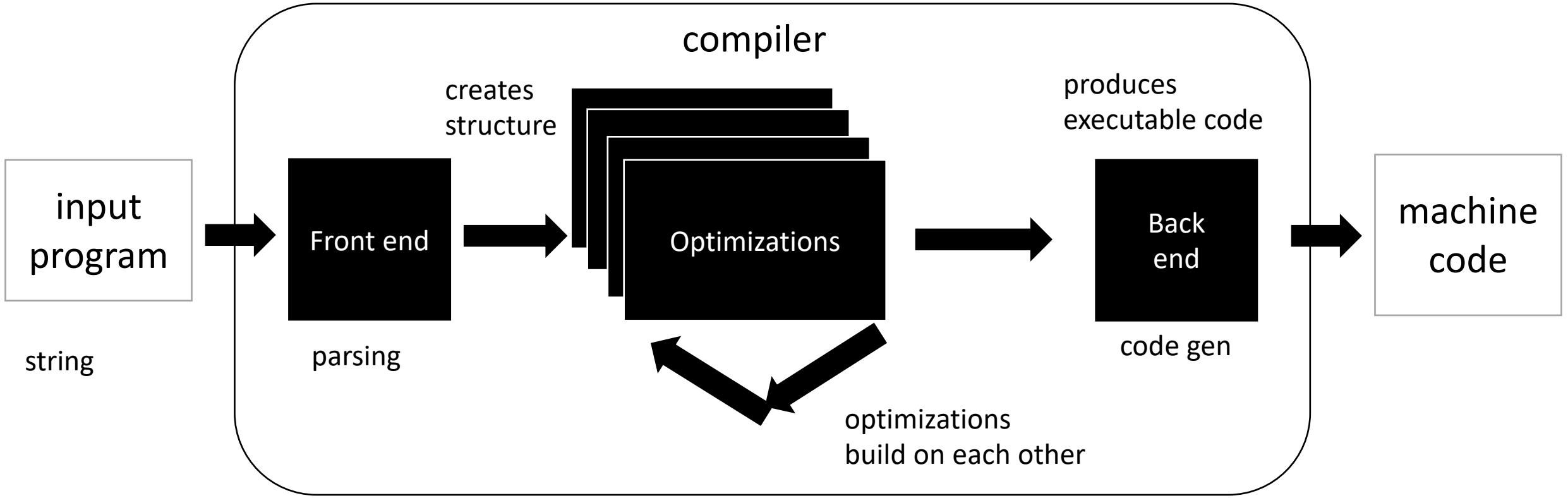
Medium detailed view

more about optimizations: https://stackoverflow.com/questions/15548023/clang-optimization-levels

Compiler Architecture

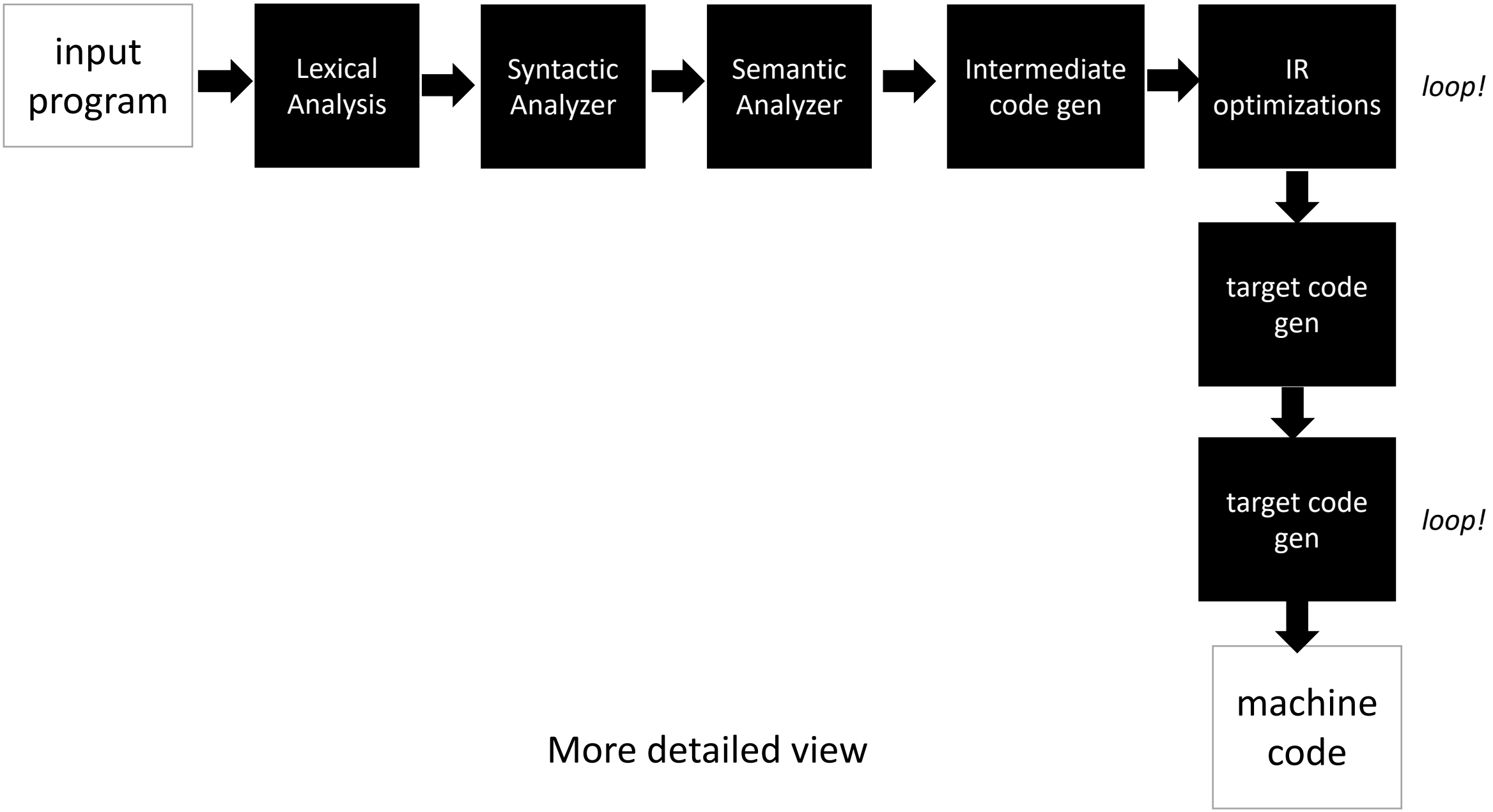
What are some of the benefits of this design?

What are some of the drawbacks of this design?

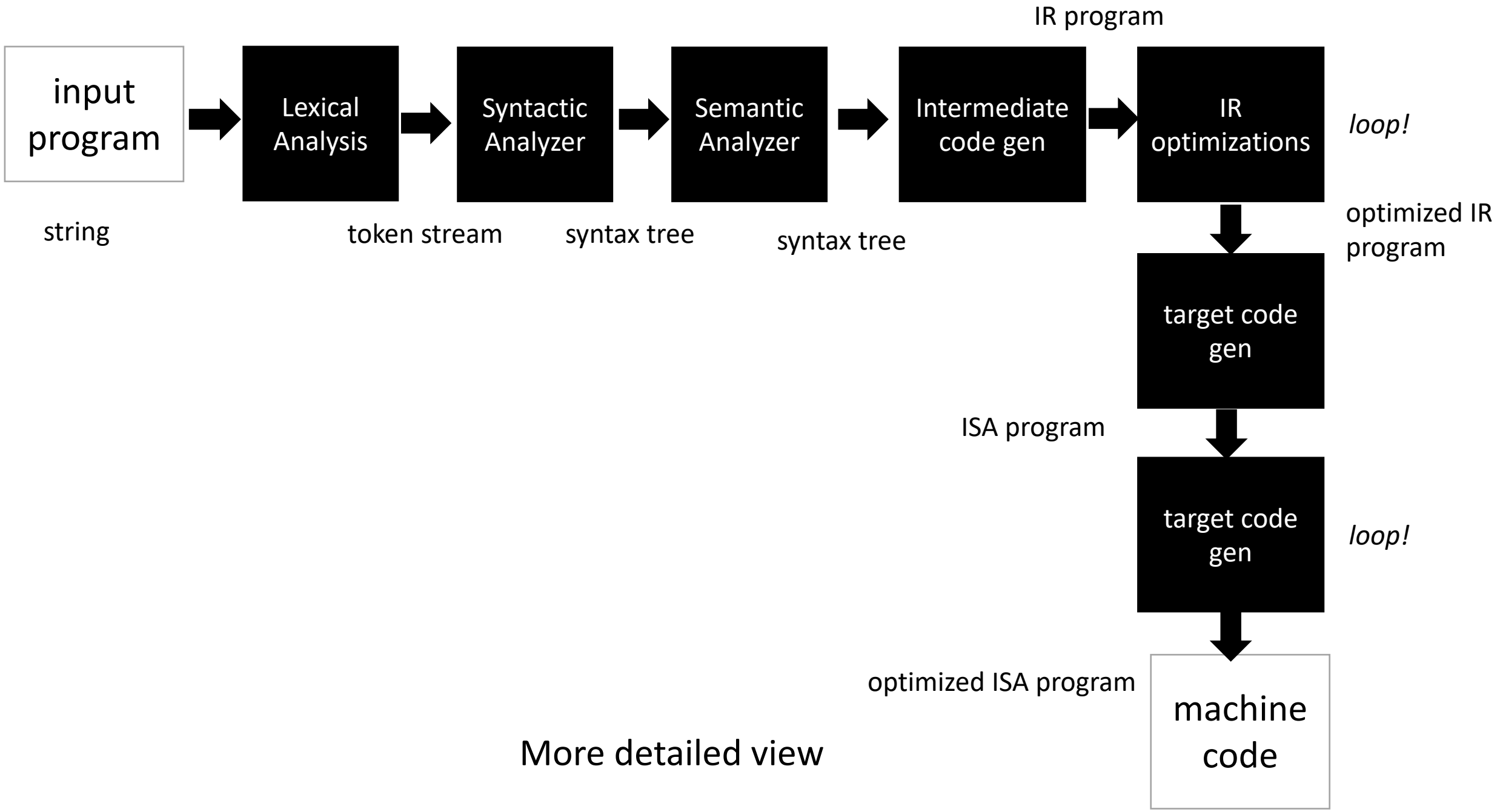


Medium detailed view

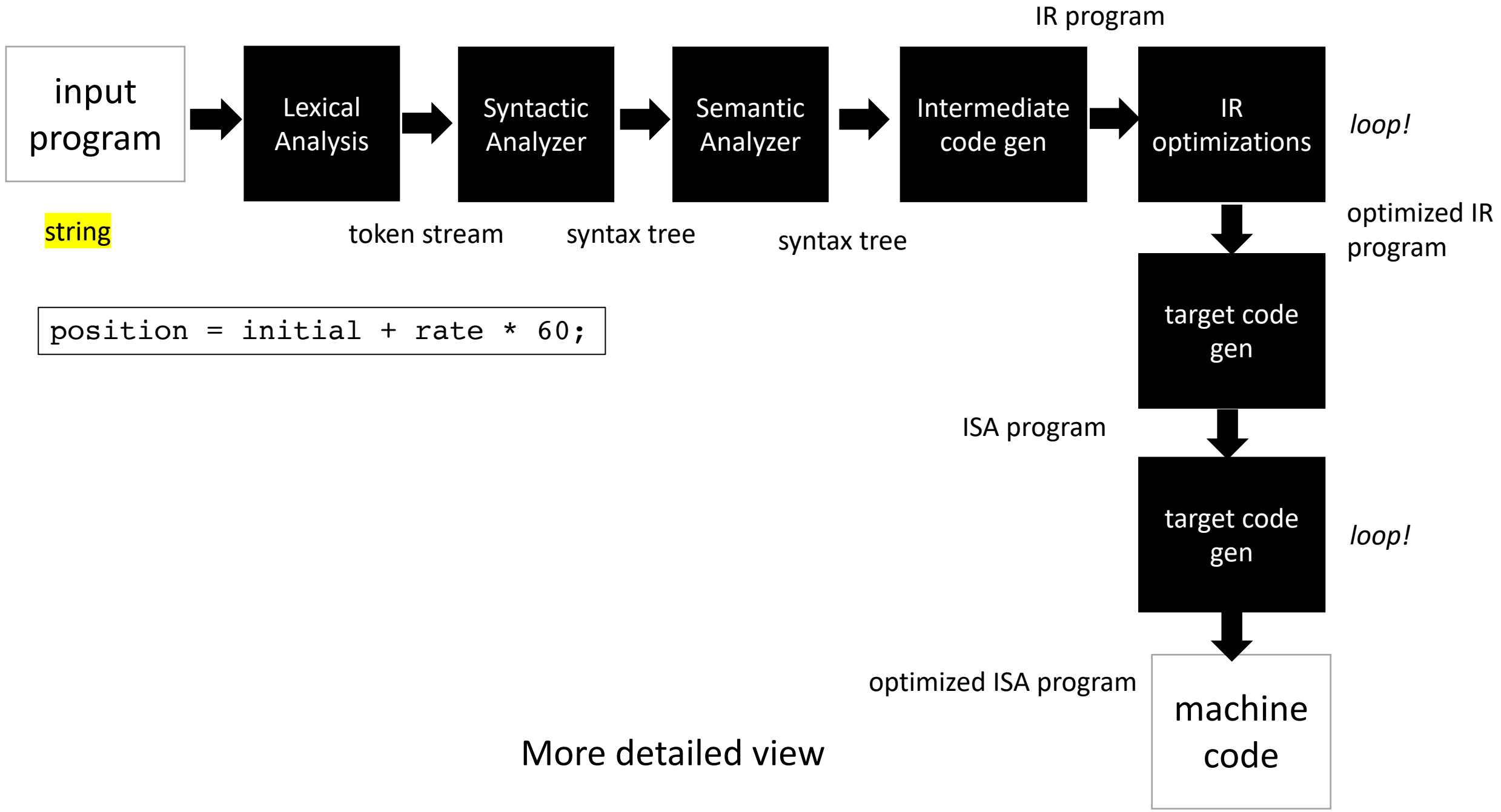
more about optimizations: https://stackoverflow.com/questions/15548023/clang-optimization-levels



More detailed view

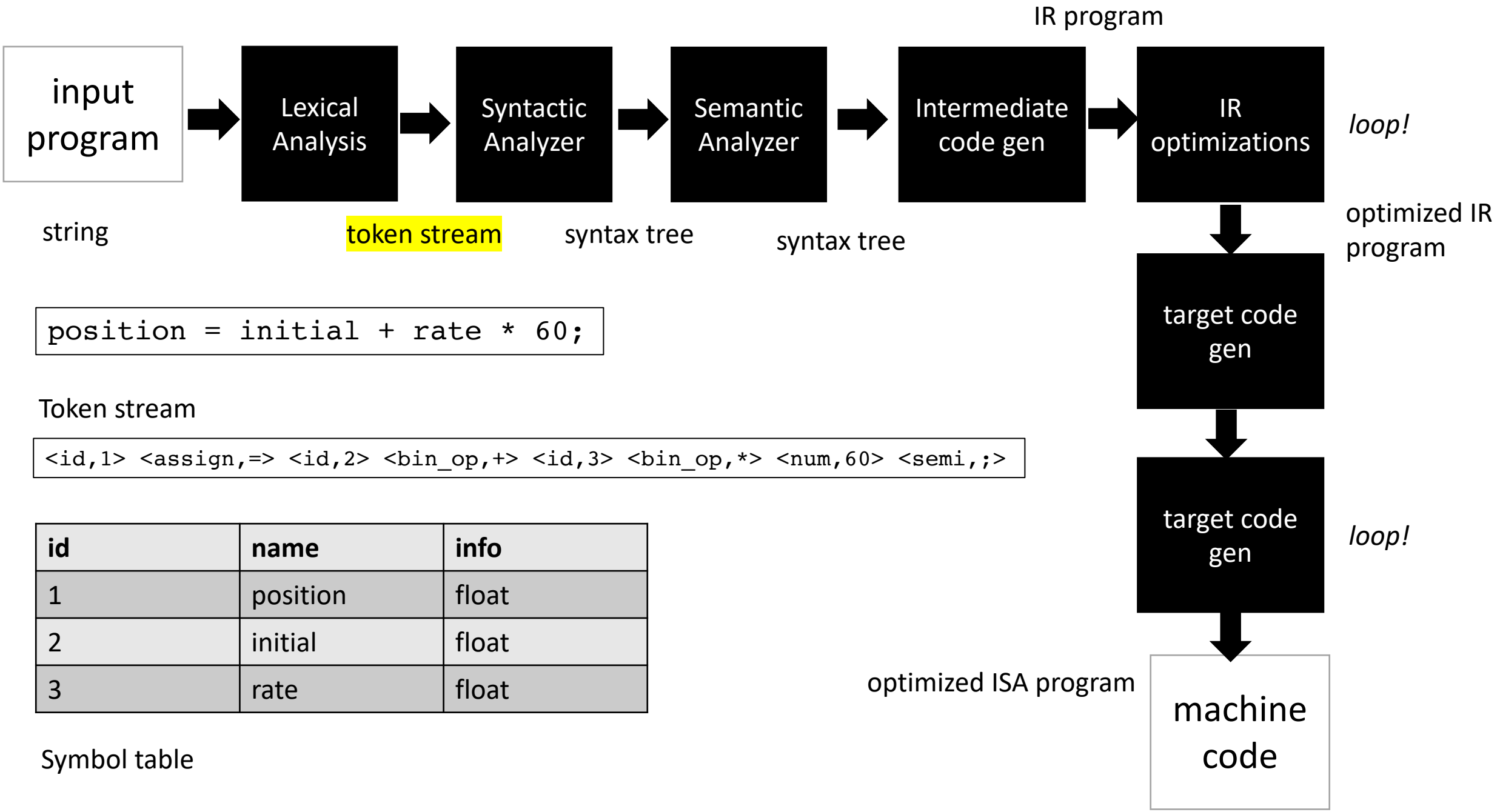


More detailed view




```
position = initial + rate * 60;
```

More detailed view




```
position = initial + rate * 60;
```

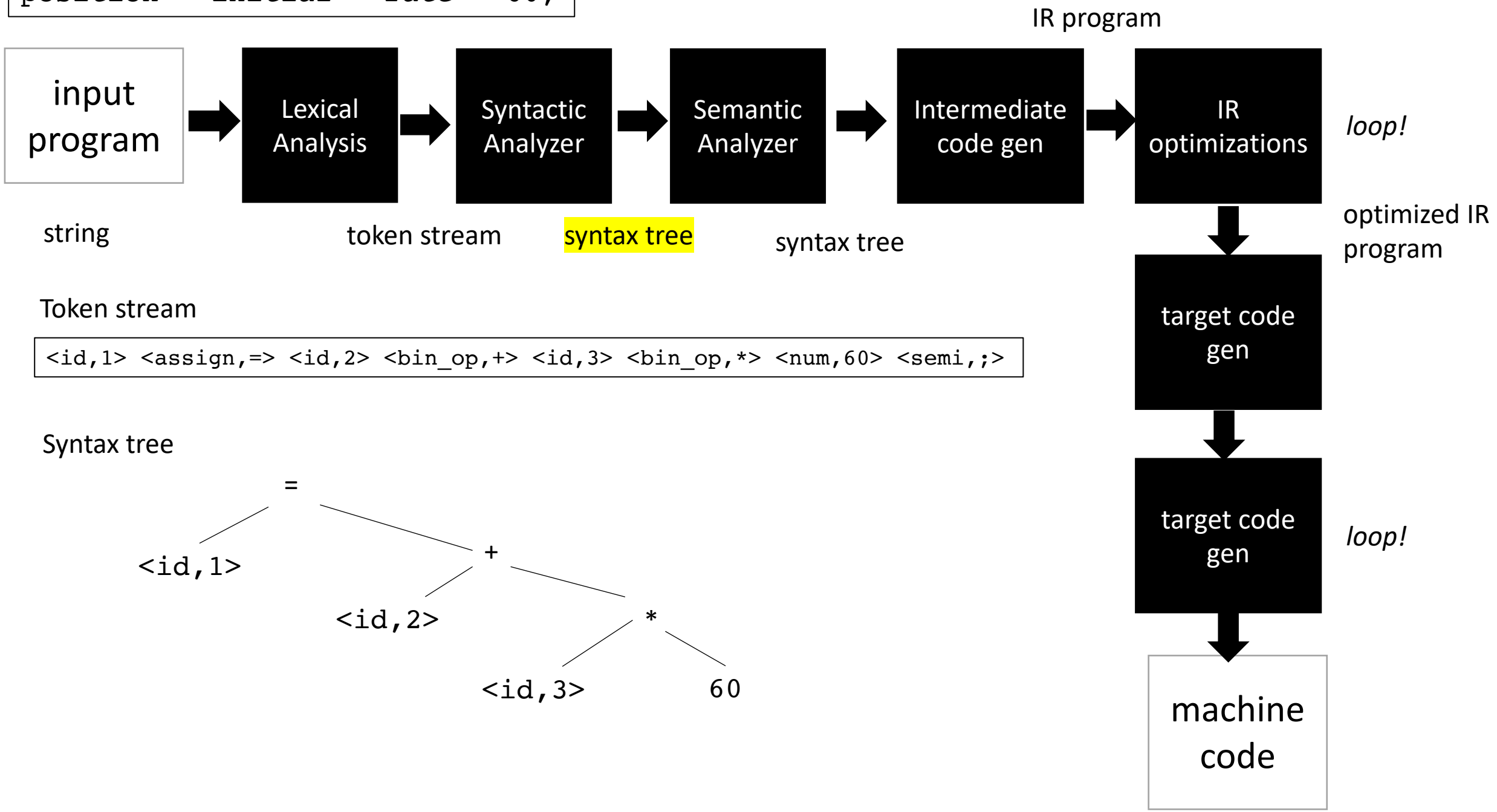
Token stream

```
<id,1> <assign,=> <id,2> <bin_op,+> <id,3> <bin_op,*> <num,60> <semi,;>
```

id	name	info
1	position	float
2	initial	float
3	rate	float

Symbol table

```
position = initial + rate * 60;
```

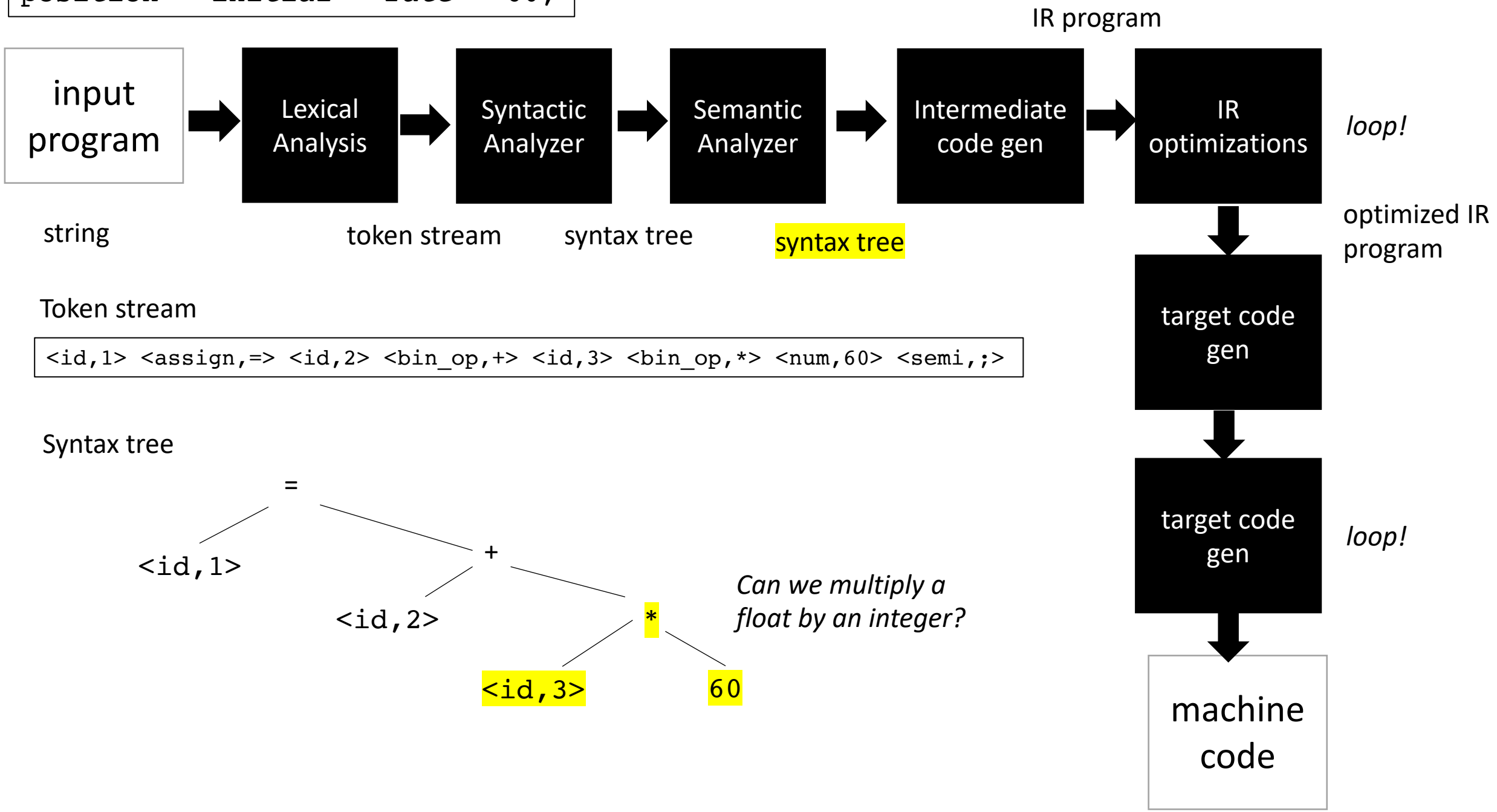


Token stream

```
<id,1> <assign,=> <id,2> <bin_op,+> <id,3> <bin_op,*> <num,60> <semi,;>
```

Syntax tree

```
position = initial + rate * 60;
```



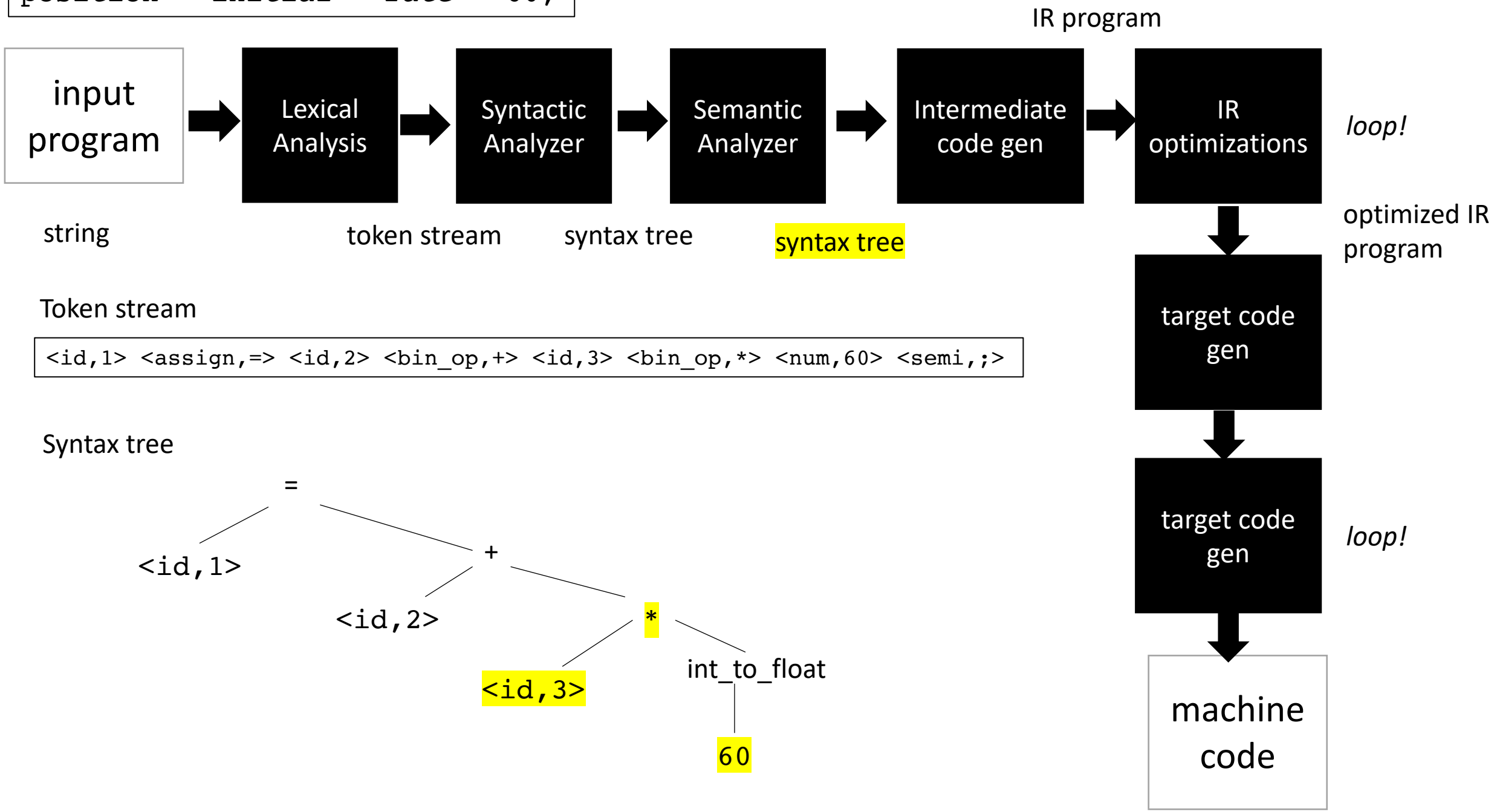
Token stream

```
<id,1> <assign,=> <id,2> <bin_op,+> <id,3> <bin_op,*> <num,60> <semi,;>
```

Syntax tree

Can we multiply a float by an integer?

```
position = initial + rate * 60;
```

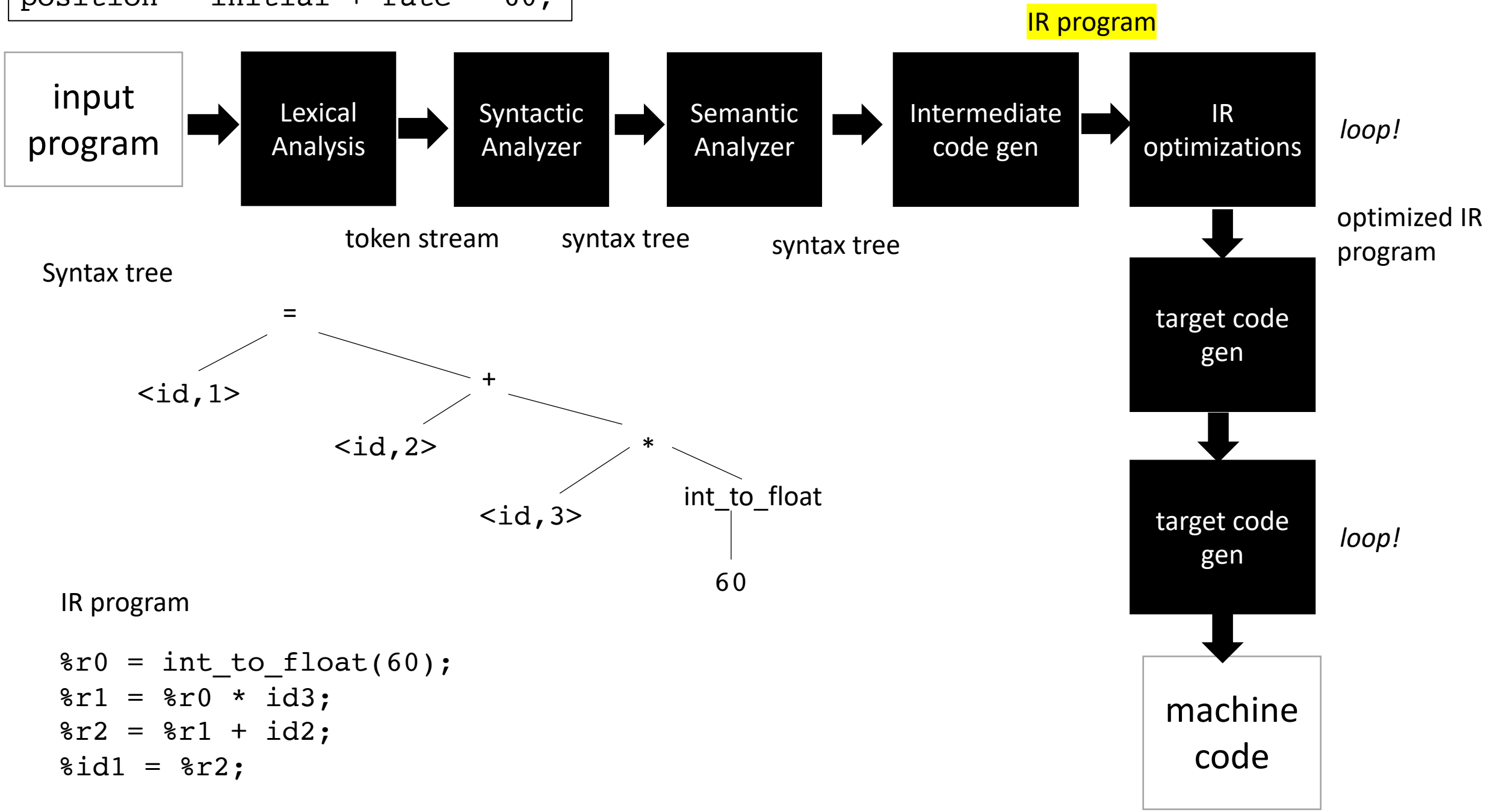


Token stream

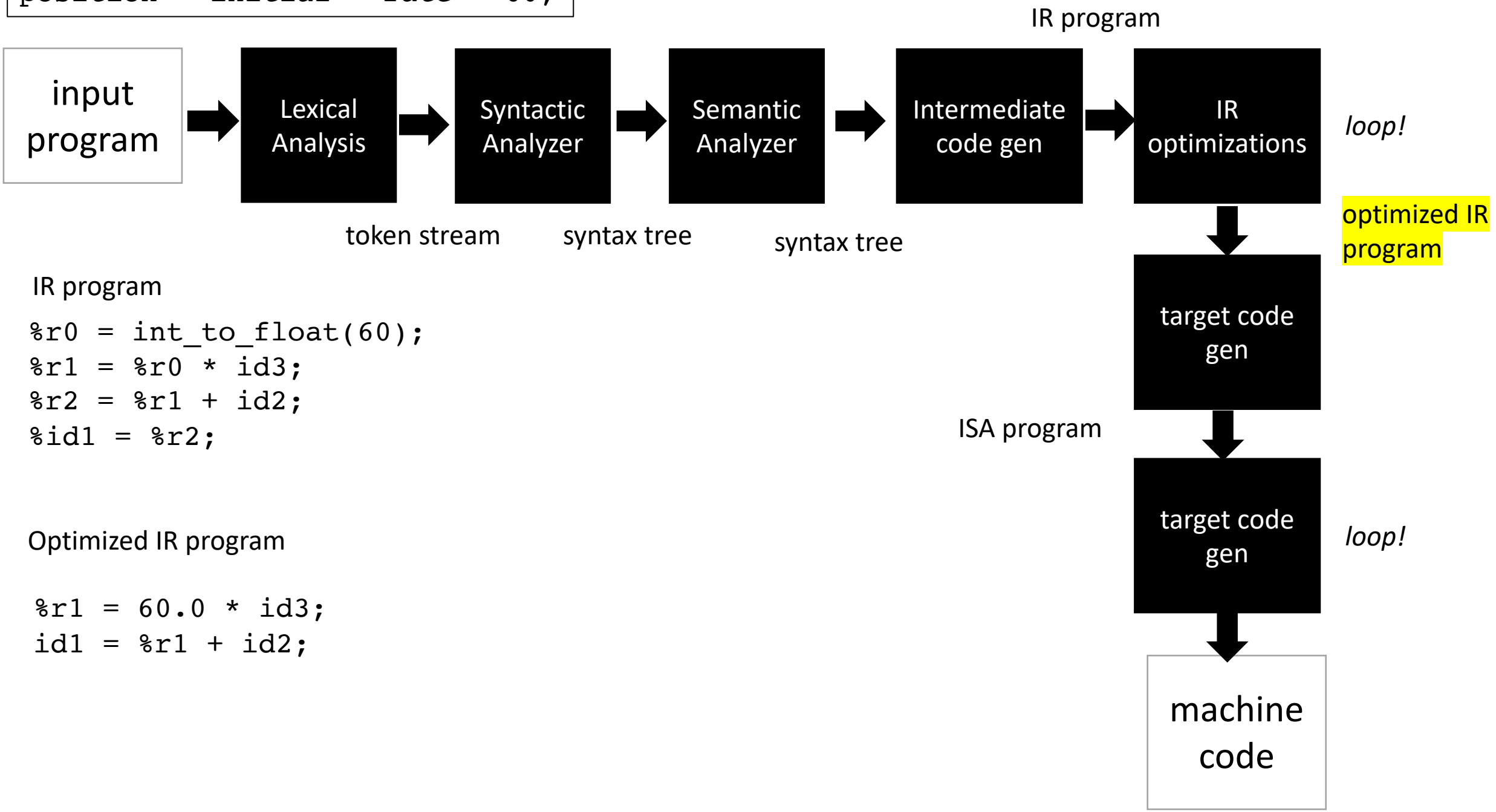
```
<id,1> <assign,=> <id,2> <bin_op,+> <id,3> <bin_op,*> <num,60> <semi,;>
```

Syntax tree

```
position = initial + rate * 60;
```



```
position = initial + rate * 60;
```



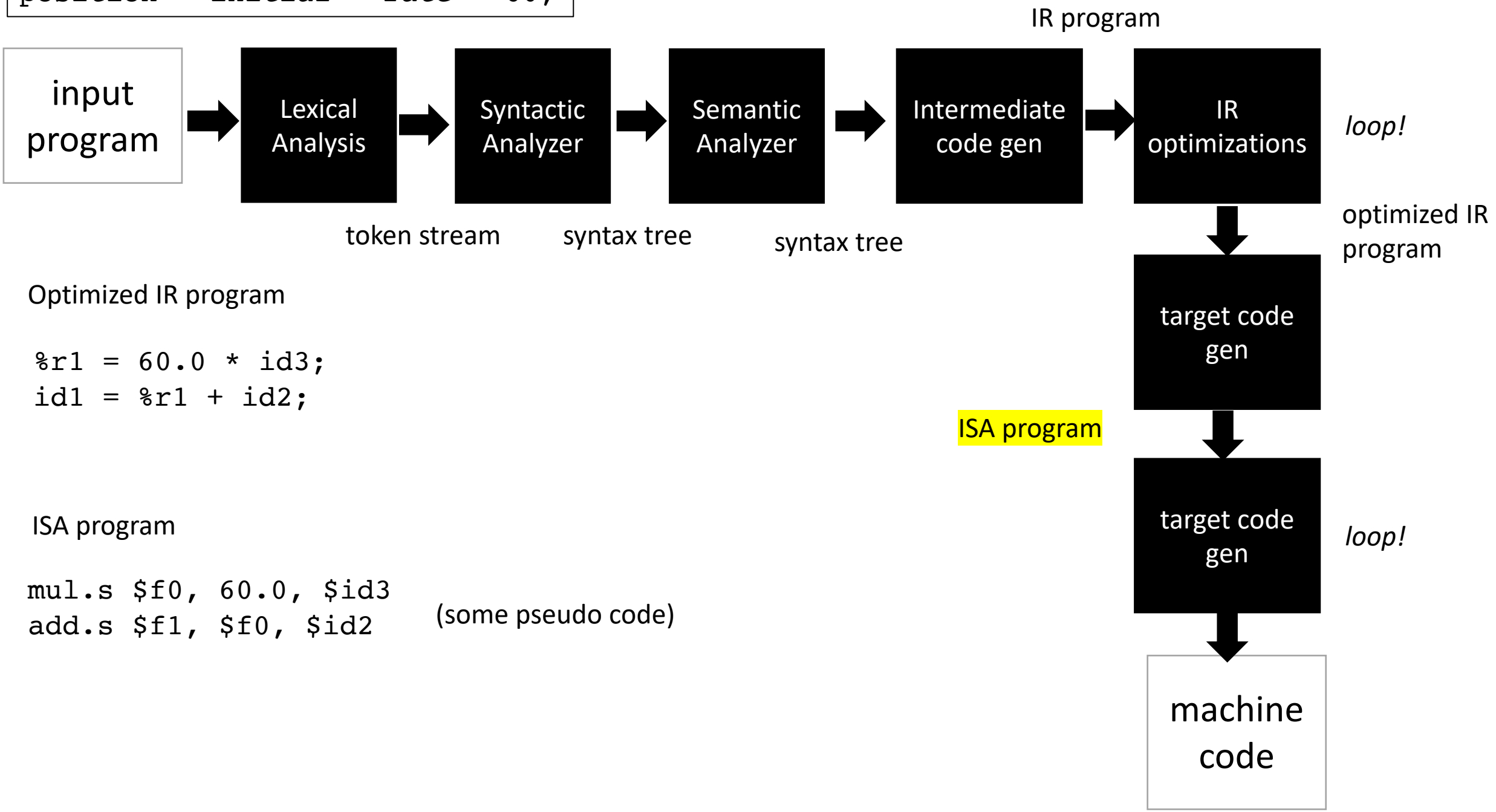
IR program

```
%r0 = int_to_float(60);
%r1 = %r0 * id3;
%r2 = %r1 + id2;
%id1 = %r2;
```

Optimized IR program

```
%r1 = 60.0 * id3;
id1 = %r1 + id2;
```

```
position = initial + rate * 60;
```



Optimized IR program

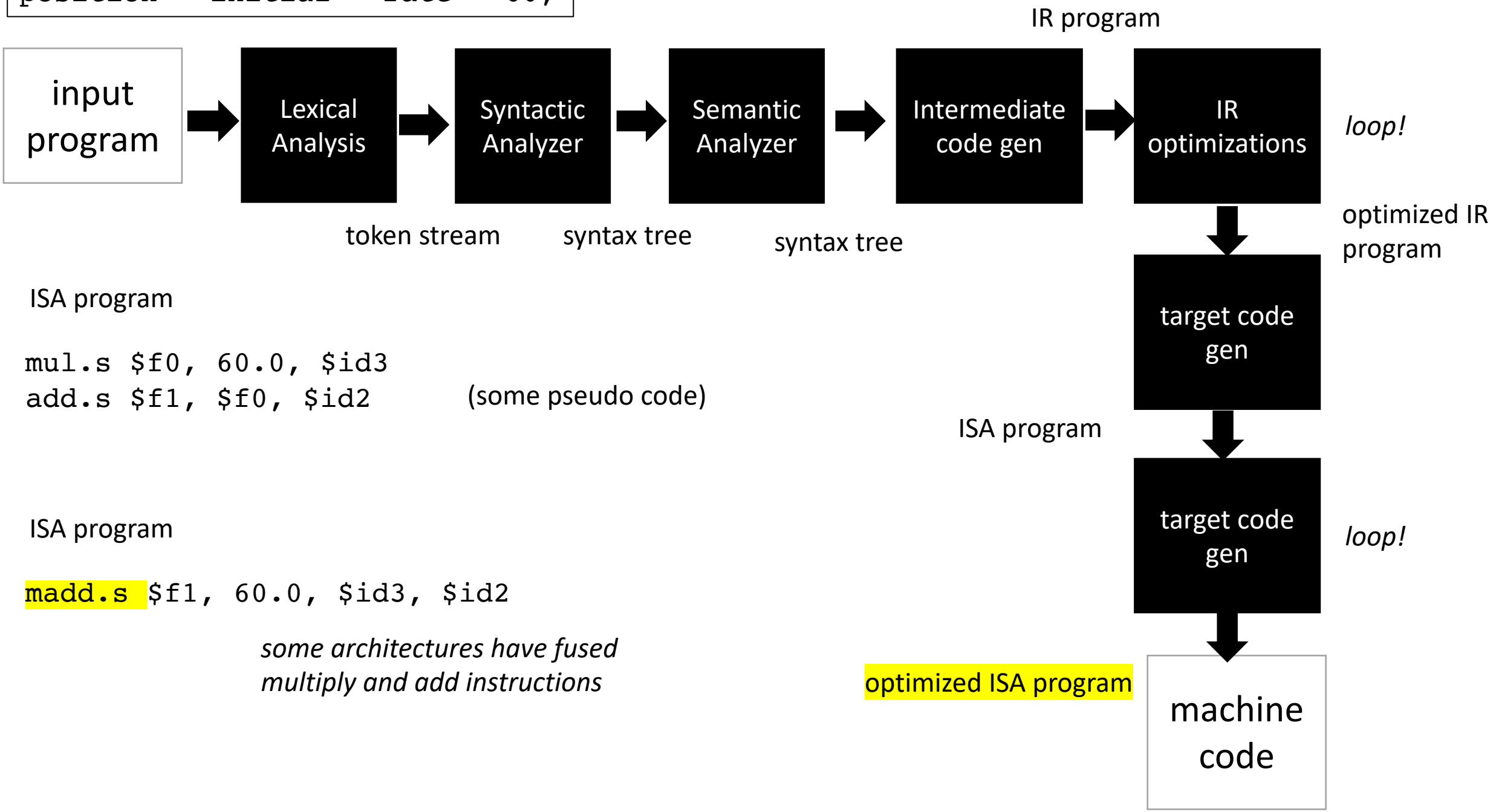
```
%r1 = 60.0 * id3;
id1 = %r1 + id2;
```

ISA program

```
mul.s $f0, 60.0, $id3
add.s $f1, $f0, $id2
```

(some pseudo code)

```
position = initial + rate * 60;
```



ISA program

```
mul.s $f0, 60.0, $id3
add.s $f1, $f0, $id2
```

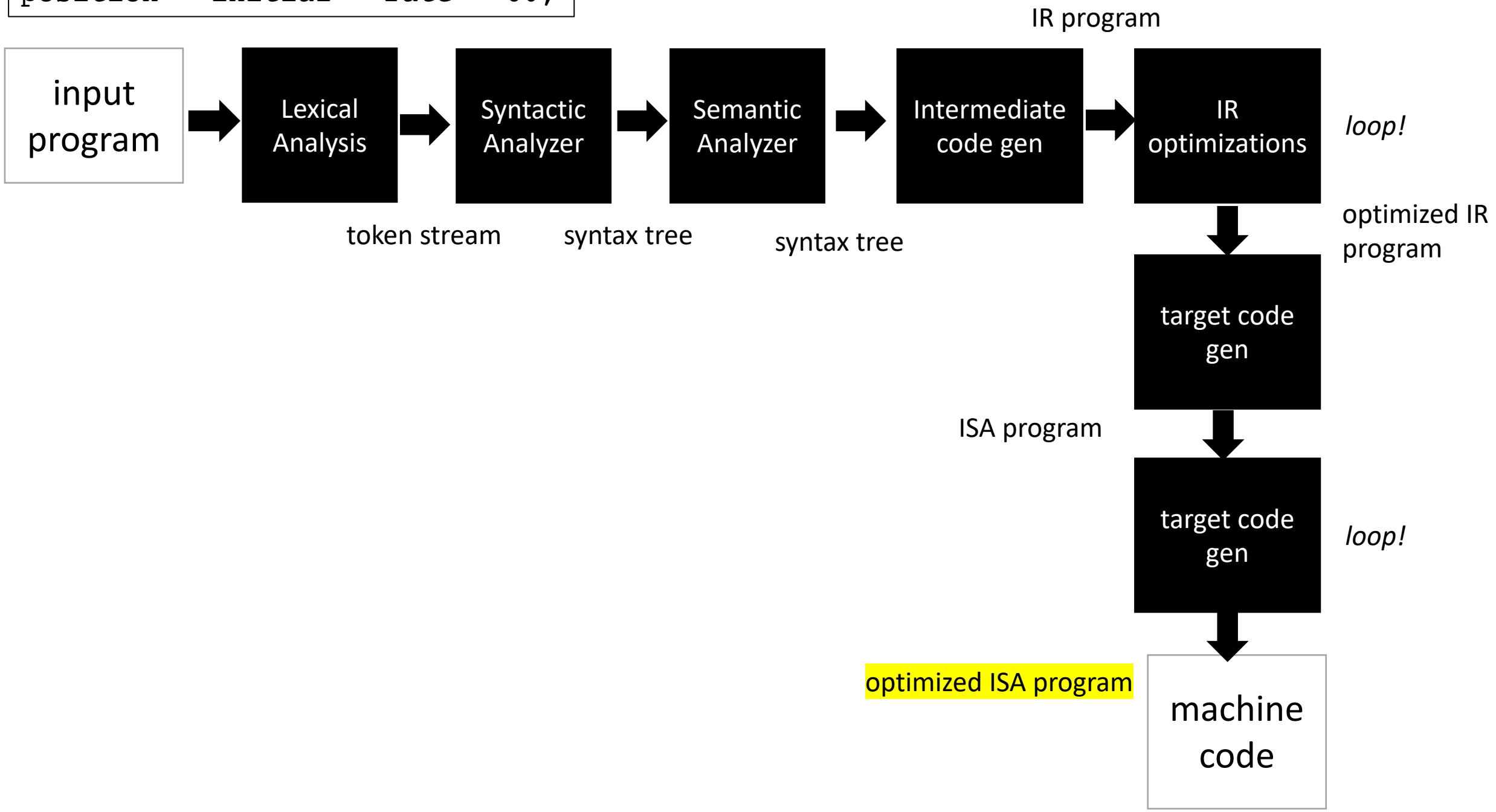
(some pseudo code)

ISA program

```
madd.s $f1, 60.0, $id3, $id2
```

some architectures have fused multiply and add instructions

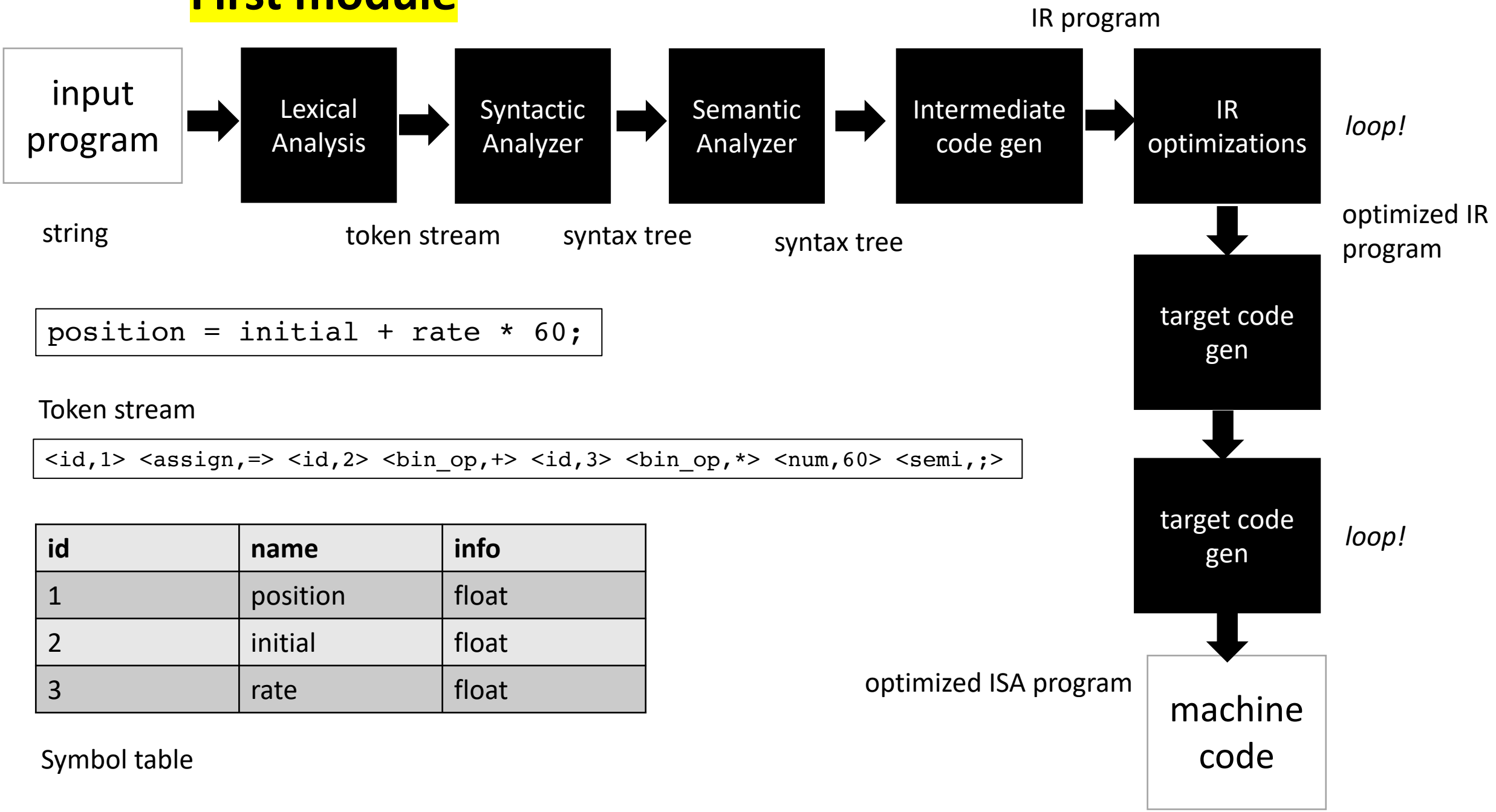
```
position = initial + rate * 60;
```



Compiler Architecture

Now you've seen a journey through a compiler!

First module




```
position = initial + rate * 60;
```

Token stream

```
<id,1> <assign,=> <id,2> <bin_op,+> <id,3> <bin_op,*> <num,60> <semi,;>
```

id	name	info
1	position	float
2	initial	float
3	rate	float

Symbol table

Next Class

- **Lexical Analysis**