

## My Project

Generated by Doxygen 1.8.13



# Contents



# Chapter 1

## Class Index

### 1.1 Class List

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

|                                     |       |    |
|-------------------------------------|-------|----|
| <a href="#">_person</a>             | ..... | ?? |
| <a href="#">node</a>                | ..... | ?? |
| <a href="#">operator_precedence</a> | ..... | ?? |
| <a href="#">ssu_scoreTable</a>      | ..... | ?? |



## Chapter 2

# File Index

### 2.1 File List

Here is a list of all files with brief descriptions:

|                               |    |
|-------------------------------|----|
| <a href="#">blank.c</a>       | ?? |
| <a href="#">blank.h</a>       | ?? |
| <a href="#">main.c</a>        | ?? |
| <a href="#">ssu_score.c</a>   | ?? |
| <a href="#">ssu_score.h</a>   | ?? |
| <a href="#">20200002/20.c</a> | ?? |
| <a href="#">20200002/21.c</a> | ?? |
| <a href="#">20200002/22.c</a> | ?? |
| <a href="#">20200002/23.c</a> | ?? |
| <a href="#">20200002/24.c</a> | ?? |
| <a href="#">20200002/25.c</a> | ?? |
| <a href="#">20200002/26.c</a> | ?? |
| <a href="#">20200002/27.c</a> | ?? |
| <a href="#">20200002/28.c</a> | ?? |
| <a href="#">20200002/29.c</a> | ?? |
| <a href="#">20200004/20.c</a> | ?? |
| <a href="#">20200004/21.c</a> | ?? |
| <a href="#">20200004/22.c</a> | ?? |
| <a href="#">20200004/23.c</a> | ?? |
| <a href="#">20200004/24.c</a> | ?? |
| <a href="#">20200004/25.c</a> | ?? |
| <a href="#">20200004/26.c</a> | ?? |
| <a href="#">20200004/27.c</a> | ?? |
| <a href="#">20200004/28.c</a> | ?? |
| <a href="#">20200004/29.c</a> | ?? |
| <a href="#">20200005/20.c</a> | ?? |
| <a href="#">20200005/21.c</a> | ?? |
| <a href="#">20200005/22.c</a> | ?? |
| <a href="#">20200005/23.c</a> | ?? |
| <a href="#">20200005/24.c</a> | ?? |
| <a href="#">20200005/25.c</a> | ?? |
| <a href="#">20200005/26.c</a> | ?? |
| <a href="#">20200005/27.c</a> | ?? |
| <a href="#">20200005/28.c</a> | ?? |
| <a href="#">20200005/29.c</a> | ?? |

|                       |    |
|-----------------------|----|
| 20200009/20.c         | ?? |
| 20200009/21.c         | ?? |
| 20200009/22.c         | ?? |
| 20200009/23.c         | ?? |
| 20200009/24.c         | ?? |
| 20200009/25.c         | ?? |
| 20200009/26.c         | ?? |
| 20200009/27.c         | ?? |
| 20200009/28.c         | ?? |
| 20200009/29.c         | ?? |
| ANS_DIR/20.c          | ?? |
| ANS_DIR/21.c          | ?? |
| ANS_DIR/22.c          | ?? |
| ANS_DIR/23.c          | ?? |
| ANS_DIR/24.c          | ?? |
| ANS_DIR/25.c          | ?? |
| ANS_DIR/26.c          | ?? |
| ANS_DIR/27.c          | ?? |
| ANS_DIR/28.c          | ?? |
| ANS_DIR/29.c          | ?? |
| STD_DIR/20200001/20.c | ?? |
| STD_DIR/20200001/21.c | ?? |
| STD_DIR/20200001/22.c | ?? |
| STD_DIR/20200001/23.c | ?? |
| STD_DIR/20200001/24.c | ?? |
| STD_DIR/20200001/25.c | ?? |
| STD_DIR/20200001/26.c | ?? |
| STD_DIR/20200001/27.c | ?? |
| STD_DIR/20200001/28.c | ?? |
| STD_DIR/20200001/29.c | ?? |
| STD_DIR/20200003/20.c | ?? |
| STD_DIR/20200003/21.c | ?? |
| STD_DIR/20200003/22.c | ?? |
| STD_DIR/20200003/23.c | ?? |
| STD_DIR/20200003/24.c | ?? |
| STD_DIR/20200003/25.c | ?? |
| STD_DIR/20200003/26.c | ?? |
| STD_DIR/20200003/27.c | ?? |
| STD_DIR/20200003/28.c | ?? |
| STD_DIR/20200003/29.c | ?? |
| STD_DIR/20200006/20.c | ?? |
| STD_DIR/20200006/21.c | ?? |
| STD_DIR/20200006/22.c | ?? |
| STD_DIR/20200006/23.c | ?? |
| STD_DIR/20200006/24.c | ?? |
| STD_DIR/20200006/25.c | ?? |
| STD_DIR/20200006/26.c | ?? |
| STD_DIR/20200006/27.c | ?? |
| STD_DIR/20200006/28.c | ?? |
| STD_DIR/20200006/29.c | ?? |
| STD_DIR/20200007/20.c | ?? |
| STD_DIR/20200007/21.c | ?? |
| STD_DIR/20200007/22.c | ?? |
| STD_DIR/20200007/23.c | ?? |
| STD_DIR/20200007/24.c | ?? |
| STD_DIR/20200007/25.c | ?? |
| STD_DIR/20200007/26.c | ?? |
| STD_DIR/20200007/27.c | ?? |



|                       |    |
|-----------------------|----|
| STD_DIR/20200007/28.c | ?? |
| STD_DIR/20200007/29.c | ?? |
| STD_DIR/20200008/20.c | ?? |
| STD_DIR/20200008/21.c | ?? |
| STD_DIR/20200008/22.c | ?? |
| STD_DIR/20200008/23.c | ?? |
| STD_DIR/20200008/24.c | ?? |
| STD_DIR/20200008/25.c | ?? |
| STD_DIR/20200008/26.c | ?? |
| STD_DIR/20200008/27.c | ?? |
| STD_DIR/20200008/28.c | ?? |
| STD_DIR/20200008/29.c | ?? |
| STD_DIR/20200010/20.c | ?? |
| STD_DIR/20200010/21.c | ?? |
| STD_DIR/20200010/22.c | ?? |
| STD_DIR/20200010/23.c | ?? |
| STD_DIR/20200010/24.c | ?? |
| STD_DIR/20200010/25.c | ?? |
| STD_DIR/20200010/26.c | ?? |
| STD_DIR/20200010/27.c | ?? |
| STD_DIR/20200010/28.c | ?? |
| STD_DIR/20200010/29.c | ?? |



## Chapter 3

# Class Documentation

### 3.1 `_person` Struct Reference

#### Public Attributes

- int [age](#)
- double [height](#)
- char [name](#) [10]

#### 3.1.1 Member Data Documentation

##### 3.1.1.1 `age`

```
int _person::age
```

##### 3.1.1.2 `height`

```
double _person::height
```

##### 3.1.1.3 `name`

```
char _person::name
```

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

- 20200002/[21.c](#)

## 3.2 node Struct Reference

```
#include <blank.h>
```

Collaboration diagram for node:

### Public Attributes

- struct [node](#) \* [child\\_head](#)
- char \* [name](#)
- struct [node](#) \* [next](#)
- struct [node](#) \* [parent](#)
- int [parentheses](#)
- struct [node](#) \* [prev](#)

### 3.2.1 Member Data Documentation

#### 3.2.1.1 child\_head

```
struct node* node::child_head
```

#### 3.2.1.2 name

```
char* node::name
```

#### 3.2.1.3 next

```
struct node* node::next
```

#### 3.2.1.4 parent

```
struct node* node::parent
```

#### 3.2.1.5 parentheses

```
int node::parentheses
```

#### 3.2.1.6 prev

```
struct node* node::prev
```

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

- [blank.h](#)

### 3.3 operator\_precedence Struct Reference

```
#include <blank.h>
```

#### Public Attributes

- char \* [operator](#)
- int [precedence](#)

#### 3.3.1 Member Data Documentation

##### 3.3.1.1 operator

```
char* operator_precedence::operator
```

##### 3.3.1.2 precedence

```
int operator_precedence::precedence
```

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

- [blank.h](#)

### 3.4 ssu\_scoreTable Struct Reference

```
#include <ssu_score.h>
```

## Public Attributes

- char [qname](#) [[FILELEN](#)]
- double [score](#)

### 3.4.1 Member Data Documentation

#### 3.4.1.1 qname

```
char ssu_scoreTable::qname[FILELEN]
```

#### 3.4.1.2 score

```
double ssu_scoreTable::score
```

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

- [ssu\\_score.h](#)

## Chapter 4

# File Documentation

### 4.1 20200002/20.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
#include <sys/stat.h>
#include <sys/types.h>
Include dependency graph for 20.c:
```

### 4.2 20200004/20.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
#include <sys/stat.h>
#include <sys/types.h>
Include dependency graph for 20.c:
```

#### Macros

- `#define BUFFER_SIZE 1024`
- `#define WORD_MAX 100`

#### Functions

- `int main (void)`

#### 4.2.1 Macro Definition Documentation

#### 4.2.1.1 BUFFER\_SIZE

```
#define BUFFER_SIZE 1024
```

#### 4.2.1.2 WORD\_MAX

```
#define WORD_MAX 100
```

### 4.2.2 Function Documentation

#### 4.2.2.1 main()

```
int main (  
    void )
```

## 4.3 20200005/20.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <fcntl.h>  
#include <sys/stat.h>  
#include <sys/types.h>  
Include dependency graph for 20.c:
```

### Macros

- `#define BUFFER_SIZE 1024`
- `#define WORD_MAX 100`

### Functions

- `int main (void)`

#### 4.3.1 Macro Definition Documentation



#### 4.3.1.1 BUFFER\_SIZE

```
#define BUFFER_SIZE 1024
```

#### 4.3.1.2 WORD\_MAX

```
#define WORD_MAX 100
```

### 4.3.2 Function Documentation

#### 4.3.2.1 main()

```
int main (  
    void )
```

## 4.4 20200009/20.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <fcntl.h>  
#include <sys/stat.h>  
#include <sys/types.h>  
Include dependency graph for 20.c:
```

### Macros

- `#define BUFFER_SIZE 1024`
- `#define WORD_MAX 100`

### Functions

- `int main (void)`

#### 4.4.1 Macro Definition Documentation

#### 4.4.1.1 BUFFER\_SIZE

```
#define BUFFER_SIZE 1024
```

#### 4.4.1.2 WORD\_MAX

```
#define WORD_MAX 100
```

### 4.4.2 Function Documentation

#### 4.4.2.1 main()

```
int main (  
    void )
```

## 4.5 ANS\_DIR/20.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <fcntl.h>  
#include <sys/stat.h>  
#include <sys/types.h>  
Include dependency graph for 20.c:
```

### Macros

- `#define` [BUFFER\\_SIZE](#) 1024
- `#define` [WORD\\_MAX](#) 100

### Functions

- `int` [main](#) (void)

#### 4.5.1 Macro Definition Documentation

#### 4.5.1.1 BUFFER\_SIZE

```
#define BUFFER_SIZE 1024
```

#### 4.5.1.2 WORD\_MAX

```
#define WORD_MAX 100
```

### 4.5.2 Function Documentation

#### 4.5.2.1 main()

```
int main (  
        void )
```

## 4.6 STD\_DIR/20200001/20.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <sys/types.h>  
Include dependency graph for 20.c:
```

### Macros

- `#define` [BUFFER\\_SIZE](#) 1024
- `#define` [WORD\\_MAX](#) 100

### Functions

- `int` [main](#) (void)

### 4.6.1 Macro Definition Documentation

#### 4.6.1.1 BUFFER\_SIZE

```
#define BUFFER_SIZE 1024
```

#### 4.6.1.2 WORD\_MAX

```
#define WORD_MAX 100
```

### 4.6.2 Function Documentation

#### 4.6.2.1 main()

```
int main (  
    void )
```

## 4.7 STD\_DIR/20200003/20.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <fcntl.h>  
#include <sys/stat.h>  
#include <sys/types.h>  
Include dependency graph for 20.c:
```

### Macros

- `#define BUFFER_SIZE 1024`
- `#define WORD_MAX 100`

### Functions

- `int main (void)`

#### 4.7.1 Macro Definition Documentation

##### 4.7.1.1 BUFFER\_SIZE

```
#define BUFFER_SIZE 1024
```

#### 4.7.1.2 WORD\_MAX

```
#define WORD_MAX 100
```

### 4.7.2 Function Documentation

#### 4.7.2.1 main()

```
int main (  
    void )
```

## 4.8 STD\_DIR/20200006/20.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <fcntl.h>  
#include <sys/stat.h>  
#include <sys/types.h>  
Include dependency graph for 20.c:
```

### Macros

- `#define BUFFER_SIZE 1024`
- `#define WORD_MAX 100`

### Functions

- `int main (void)`

### 4.8.1 Macro Definition Documentation

#### 4.8.1.1 BUFFER\_SIZE

```
#define BUFFER_SIZE 1024
```

#### 4.8.1.2 WORD\_MAX

```
#define WORD_MAX 100
```

### 4.8.2 Function Documentation

#### 4.8.2.1 main()

```
int main (  
    void )
```

## 4.9 STD\_DIR/20200007/20.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <fcntl.h>  
#include <sys/stat.h>  
#include <sys/types.h>  
Include dependency graph for 20.c:
```

### Macros

- `#define BUFFER_SIZE 1024`
- `#define WORD_MAX 100`

### Functions

- `int main (void)`

### 4.9.1 Macro Definition Documentation

#### 4.9.1.1 BUFFER\_SIZE

```
#define BUFFER_SIZE 1024
```

#### 4.9.1.2 WORD\_MAX

```
#define WORD_MAX 100
```

### 4.9.2 Function Documentation

#### 4.9.2.1 main()

```
int main (  
    void )
```

## 4.10 STD\_DIR/20200008/20.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <fcntl.h>  
#include <sys/stat.h>  
#include <sys/types.h>  
Include dependency graph for 20.c:
```

### Macros

- `#define BUFFER_SIZE 1024`
- `#define WORD_MAX 100`

### Functions

- `int main (void)`

#### 4.10.1 Macro Definition Documentation

##### 4.10.1.1 BUFFER\_SIZE

```
#define BUFFER_SIZE 1024
```

#### 4.10.1.2 WORD\_MAX

```
#define WORD_MAX 100
```

### 4.10.2 Function Documentation

#### 4.10.2.1 main()

```
int main (  
    void )
```

## 4.11 STD\_DIR/20200010/20.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <fcntl.h>  
#include <sys/stat.h>  
#include <sys/types.h>  
Include dependency graph for 20.c:
```

### Macros

- `#define BUFFER_SIZE 1024`
- `#define WORD_MAX 100`

### Functions

- `int main (void)`

#### 4.11.1 Macro Definition Documentation

##### 4.11.1.1 BUFFER\_SIZE

```
#define BUFFER_SIZE 1024
```



#### 4.11.1.2 WORD\_MAX

```
#define WORD_MAX 100
```

### 4.11.2 Function Documentation

#### 4.11.2.1 main()

```
int main (  
    void )
```

## 4.12 20200002/21.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
Include dependency graph for 21.c:
```

### Classes

- struct [\\_person](#)

### Typedefs

- typedef struct [\\_person](#) [Person](#)

### Functions

- int [main](#) (void)

### 4.12.1 Typedef Documentation

#### 4.12.1.1 Person

```
typedef struct \_person Person
```

### 4.12.2 Function Documentation

#### 4.12.2.1 main()

```
int main (  
    void )
```

### 4.13 20200004/21.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
Include dependency graph for 21.c:
```

#### Classes

- struct [\\_person](#)

#### Typedefs

- typedef struct [\\_person](#) [Person](#)

#### Functions

- int [main](#) (void)

#### 4.13.1 Typedef Documentation

##### 4.13.1.1 Person

```
typedef struct \_person Person
```

#### 4.13.2 Function Documentation

##### 4.13.2.1 main()

```
int main (  
    void )
```

## 4.14 20200005/21.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
Include dependency graph for 21.c:
```

### Classes

- struct [\\_person](#)

### Typedefs

- typedef struct [\\_person](#) [Person](#)

### Functions

- int [main](#) (void)

#### 4.14.1 Typedef Documentation

##### 4.14.1.1 Person

```
typedef struct \_person Person
```

#### 4.14.2 Function Documentation

##### 4.14.2.1 main()

```
int main (
    void )
```

## 4.15 20200009/21.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
Include dependency graph for 21.c:
```

## Classes

- struct [\\_person](#)

## Typedefs

- typedef struct [\\_person](#) [Person](#)

## Functions

- int [main](#) (void)

### 4.15.1 Typedef Documentation

#### 4.15.1.1 Person

```
typedef struct \_person Person
```

### 4.15.2 Function Documentation

#### 4.15.2.1 main()

```
int main (  
    void )
```

## 4.16 ANS\_DIR/21.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
Include dependency graph for 21.c:
```

## Classes

- struct [\\_person](#)

## Typedefs

- typedef struct [\\_person](#) [Person](#)

## Functions

- int [main](#) (void)

### 4.16.1 Typedef Documentation

#### 4.16.1.1 Person

```
typedef struct \_person Person
```

### 4.16.2 Function Documentation

#### 4.16.2.1 main()

```
int main (  
    void )
```

## 4.17 STD\_DIR/20200001/21.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
Include dependency graph for 21.c:
```

## Classes

- struct [\\_person](#)

## Typedefs

- typedef struct [\\_person](#) Person

## Functions

- int [main](#) (void)

### 4.17.1 Typedef Documentation

#### 4.17.1.1 Person

```
typedef struct _person Person
```

### 4.17.2 Function Documentation

#### 4.17.2.1 main()

```
int main (  
        void )
```

## 4.18 STD\_DIR/20200003/21.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
Include dependency graph for 21.c:
```

### Classes

- struct [\\_person](#)

### Typedefs

- typedef struct [\\_person](#) Person

### Functions

- int [main](#) (void)

### 4.18.1 Typedef Documentation

#### 4.18.1.1 Person

```
typedef struct _person Person
```

### 4.18.2 Function Documentation

#### 4.18.2.1 main()

```
int main (  
    void )
```

## 4.19 STD\_DIR/20200006/21.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
Include dependency graph for 21.c:
```

### Classes

- struct [\\_person](#)

### Typedefs

- typedef struct [\\_person](#) [Person](#)

### Functions

- int [main](#) (void)

## 4.19.1 Typedef Documentation

### 4.19.1.1 Person

```
typedef struct \_person Person
```

## 4.19.2 Function Documentation

### 4.19.2.1 main()

```
int main (  
    void )
```

## 4.20 STD\_DIR/20200007/21.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
Include dependency graph for 21.c:
```

### Classes

- struct [\\_person](#)

### Typedefs

- typedef struct [\\_person](#) [Person](#)

### Functions

- int [main](#) (void)

#### 4.20.1 Typedef Documentation

##### 4.20.1.1 Person

```
typedef struct \_person Person
```

#### 4.20.2 Function Documentation

##### 4.20.2.1 main()

```
int main (
    void )
```

## 4.21 STD\_DIR/20200008/21.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
Include dependency graph for 21.c:
```



## Classes

- struct [\\_person](#)

## Typedefs

- typedef struct [\\_person](#) [Person](#)

## Functions

- int [main](#) (void)

### 4.21.1 Typedef Documentation

#### 4.21.1.1 Person

```
typedef struct \_person Person
```

### 4.21.2 Function Documentation

#### 4.21.2.1 main()

```
int main (  
    void )
```

## 4.22 STD\_DIR/20200010/21.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
Include dependency graph for 21.c:
```

## Classes

- struct [\\_person](#)

## Typedefs

- typedef struct [\\_person](#) [Person](#)

## Functions

- int `main` (void)

### 4.22.1 Typedef Documentation

#### 4.22.1.1 Person

```
typedef struct _person Person
```

### 4.22.2 Function Documentation

#### 4.22.2.1 main()

```
int main (  
    void )
```

## 4.23 20200002/22.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <fcntl.h>  
Include dependency graph for 22.c:
```

## Macros

- #define `BUFFER_SIZE` 1024

## Functions

- int `main` (void)

### 4.23.1 Macro Definition Documentation

#### 4.23.1.1 BUFFER\_SIZE

```
#define BUFFER_SIZE 1024
```

### 4.23.2 Function Documentation

#### 4.23.2.1 main()

```
int main (  
    void )
```

## 4.24 20200004/22.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <fcntl.h>  
Include dependency graph for 22.c:
```

### Macros

- #define [BUFFER\\_SIZE](#) 1024

### Functions

- int [main](#) (void)

### 4.24.1 Macro Definition Documentation

#### 4.24.1.1 BUFFER\_SIZE

```
#define BUFFER_SIZE 1024
```

### 4.24.2 Function Documentation

#### 4.24.2.1 main()

```
int main (  
    void )
```

### 4.25 20200005/22.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <fcntl.h>  
Include dependency graph for 22.c:
```

#### Macros

- #define [BUFFER\\_SIZE](#) 1024

#### Functions

- int [main](#) (void)

#### 4.25.1 Macro Definition Documentation

##### 4.25.1.1 BUFFER\_SIZE

```
#define BUFFER_SIZE 1024
```

#### 4.25.2 Function Documentation

##### 4.25.2.1 main()

```
int main (  
    void )
```

### 4.26 20200009/22.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <fcntl.h>  
Include dependency graph for 22.c:
```

## Macros

- #define `BUFFER_SIZE` 1024

## Functions

- int `main` (void)

### 4.26.1 Macro Definition Documentation

#### 4.26.1.1 `BUFFER_SIZE`

```
#define BUFFER_SIZE 1024
```

### 4.26.2 Function Documentation

#### 4.26.2.1 `main()`

```
int main (  
    void )
```

## 4.27 ANS\_DIR/22.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <fcntl.h>  
Include dependency graph for 22.c:
```

## Macros

- #define `BUFFER_SIZE` 1024

## Functions

- int `main` (void)

## 4.27.1 Macro Definition Documentation

### 4.27.1.1 BUFFER\_SIZE

```
#define BUFFER_SIZE 1024
```

## 4.27.2 Function Documentation

### 4.27.2.1 main()

```
int main (  
    void )
```

## 4.28 STD\_DIR/20200001/22.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <fcntl.h>  
Include dependency graph for 22.c:
```

### Macros

- `#define` [BUFFER\\_SIZE](#) 1024

### Functions

- `int` [main](#) (void)

## 4.28.1 Macro Definition Documentation

### 4.28.1.1 BUFFER\_SIZE

```
#define BUFFER_SIZE 1024
```

## 4.28.2 Function Documentation

### 4.28.2.1 main()

```
int main (  
    void )
```

## 4.29 STD\_DIR/20200003/22.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <fcntl.h>  
Include dependency graph for 22.c:
```

### Macros

- `#define BUFFER_SIZE 1024`

### Functions

- `int main (void)`

## 4.29.1 Macro Definition Documentation

### 4.29.1.1 BUFFER\_SIZE

```
#define BUFFER_SIZE 1024
```

## 4.29.2 Function Documentation

### 4.29.2.1 main()

```
int main (  
    void )
```

## 4.30 STD\_DIR/20200006/22.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
Include dependency graph for 22.c:
```

### Macros

- #define [BUFFER\\_SIZE](#) 1024

### Functions

- int [main](#) (void)

### 4.30.1 Macro Definition Documentation

#### 4.30.1.1 BUFFER\_SIZE

```
#define BUFFER_SIZE 1024
```

### 4.30.2 Function Documentation

#### 4.30.2.1 main()

```
int main (
    void )
```

## 4.31 STD\_DIR/20200007/22.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
Include dependency graph for 22.c:
```

### Macros

- #define [BUFFER\\_SIZE](#) 1024



## Functions

- int [main](#) (void)

### 4.31.1 Macro Definition Documentation

#### 4.31.1.1 BUFFER\_SIZE

```
#define BUFFER_SIZE 1024
```

### 4.31.2 Function Documentation

#### 4.31.2.1 main()

```
int main (  
    void )
```

## 4.32 STD\_DIR/20200008/22.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <fcntl.h>  
Include dependency graph for 22.c:
```

## Macros

- #define [BUFFER\\_SIZE](#) 1024

## Functions

- int [main](#) (void)

### 4.32.1 Macro Definition Documentation

#### 4.32.1.1 BUFFER\_SIZE

```
#define BUFFER_SIZE 1024
```

### 4.32.2 Function Documentation

#### 4.32.2.1 main()

```
int main (  
    void )
```

## 4.33 STD\_DIR/20200010/22.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <fcntl.h>  
Include dependency graph for 22.c:
```

### Macros

- `#define` [BUFFER\\_SIZE](#) 1024

### Functions

- `int` [main](#) (void)

### 4.33.1 Macro Definition Documentation

#### 4.33.1.1 BUFFER\_SIZE

```
#define BUFFER_SIZE 1024
```

### 4.33.2 Function Documentation

#### 4.33.2.1 main()

```
int main (  
    void )
```

## 4.34 20200002/23.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <fcntl.h>  
#include <string.h>  
Include dependency graph for 23.c:
```

### Functions

- int [main](#) ()
- int [ssu\\_write](#) (int fd, char \*buf)

#### 4.34.1 Function Documentation

##### 4.34.1.1 main()

```
int main (  
    void )
```

Here is the call graph for this function:

##### 4.34.1.2 ssu\_write()

```
int ssu_write (  
    int fd,  
    char * buf )
```

Here is the caller graph for this function:

## 4.35 20200004/23.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <fcntl.h>  
#include <string.h>  
Include dependency graph for 23.c:
```

## Functions

- int [main](#) ()
- int [ssu\\_write](#) (int fd, char \*buf)

### 4.35.1 Function Documentation

#### 4.35.1.1 main()

```
int main (  
    void )
```

Here is the call graph for this function:

#### 4.35.1.2 ssu\_write()

```
int ssu_write (  
    int fd,  
    char * buf )
```

## 4.36 20200005/23.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <string.h>  
Include dependency graph for 23.c:
```

## Functions

- int [main](#) ()
- int [ssu\\_write](#) (int fd, char \*buf)

### 4.36.1 Function Documentation

#### 4.36.1.1 main()

```
int main (  
    void )
```

Here is the call graph for this function:

#### 4.36.1.2 ssu\_write()

```
int ssu_write (
    int fd,
    char * buf )
```

## 4.37 20200009/23.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
#include <string.h>
Include dependency graph for 23.c:
```

### Functions

- int [main](#) ()
- int [ssu\\_write](#) (int fd, char \*buf)

#### 4.37.1 Function Documentation

##### 4.37.1.1 main()

```
int main (
    void )
```

Here is the call graph for this function:

##### 4.37.1.2 ssu\_write()

```
int ssu_write (
    int fd,
    char * buf )
```

## 4.38 ANS\_DIR/23.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
#include <string.h>
Include dependency graph for 23.c:
```

## Functions

- int [main](#) ()
- int [ssu\\_write](#) (int fd, char \*buf)

### 4.38.1 Function Documentation

#### 4.38.1.1 main()

```
int main (  
    void )
```

Here is the call graph for this function:

#### 4.38.1.2 ssu\_write()

```
int ssu_write (  
    int fd,  
    char * buf )
```

## 4.39 STD\_DIR/20200001/23.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <fcntl.h>  
#include <string.h>  
Include dependency graph for 23.c:
```

## Functions

- int [main](#) ()
- int [ssu\\_write](#) (int fd, char \*buf)

### 4.39.1 Function Documentation

#### 4.39.1.1 main()

```
int main (  
    void )
```

Here is the call graph for this function:

#### 4.39.1.2 ssu\_write()

```
int ssu_write (
    int fd,
    char * buf )
```

## 4.40 STD\_DIR/20200003/23.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
#include <string.h>
Include dependency graph for 23.c:
```

### Functions

- int [main](#) ()
- int [ssu\\_write](#) (int fd, char \*buf)

#### 4.40.1 Function Documentation

##### 4.40.1.1 main()

```
int main (
    void )
```

Here is the call graph for this function:

##### 4.40.1.2 ssu\_write()

```
int ssu_write (
    int fd,
    char * buf )
```

## 4.41 STD\_DIR/20200006/23.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
#include <string.h>
Include dependency graph for 23.c:
```

## Functions

- int [main](#) ()
- int [ssu\\_write](#) (int fd, char \*buf)

### 4.41.1 Function Documentation

#### 4.41.1.1 main()

```
int main (  
    void )
```

Here is the call graph for this function:

#### 4.41.1.2 ssu\_write()

```
int ssu_write (  
    int fd,  
    char * buf )
```

## 4.42 STD\_DIR/20200007/23.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <fcntl.h>  
#include <string.h>  
Include dependency graph for 23.c:
```

## Functions

- int [main](#) ()
- int [ssu\\_write](#) (int fd, char \*buf)

### 4.42.1 Function Documentation

#### 4.42.1.1 main()

```
int main (  
    void )
```

Here is the call graph for this function:



#### 4.42.1.2 ssu\_write()

```
int ssu_write (
    int fd,
    char * buf )
```

### 4.43 STD\_DIR/20200008/23.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
#include <string.h>
Include dependency graph for 23.c:
```

#### Functions

- int [main](#) ()
- int [ssu\\_write](#) (int fd, char \*buf)

#### 4.43.1 Function Documentation

##### 4.43.1.1 main()

```
int main (
    void )
```

Here is the call graph for this function:

##### 4.43.1.2 ssu\_write()

```
int ssu_write (
    int fd,
    char * buf )
```

### 4.44 STD\_DIR/20200010/23.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
#include <string.h>
Include dependency graph for 23.c:
```

## Functions

- int `main` ()
- int `ssu_write` (int fd, char \*buf)

### 4.44.1 Function Documentation

#### 4.44.1.1 `main()`

```
int main (  
    void )
```

Here is the call graph for this function:

#### 4.44.1.2 `ssu_write()`

```
int ssu_write (  
    int fd,  
    char * buf )
```

## 4.45 20200002/24.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <fcntl.h>  
Include dependency graph for 24.c:
```

## Functions

- void `main` ()

### 4.45.1 Function Documentation

#### 4.45.1.1 `main()`

```
void main (  
    void )
```

## 4.46 20200004/24.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
Include dependency graph for 24.c:
```

### Functions

- void `main` ()

#### 4.46.1 Function Documentation

##### 4.46.1.1 `main()`

```
void main (
    void )
```

## 4.47 20200005/24.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
Include dependency graph for 24.c:
```

### Functions

- void `main` ()

#### 4.47.1 Function Documentation

##### 4.47.1.1 `main()`

```
void main (
    void )
```

## 4.48 20200009/24.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
Include dependency graph for 24.c:
```

### Functions

- void [main](#) ()

#### 4.48.1 Function Documentation

##### 4.48.1.1 main()

```
void main (
    void )
```

## 4.49 ANS\_DIR/24.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
Include dependency graph for 24.c:
```

### Functions

- void [main](#) ()

#### 4.49.1 Function Documentation

##### 4.49.1.1 main()

```
void main (
    void )
```

## 4.50 STD\_DIR/20200001/24.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
Include dependency graph for 24.c:
```

### Functions

- void [main](#) ()

#### 4.50.1 Function Documentation

##### 4.50.1.1 main()

```
void main (
    void )
```

## 4.51 STD\_DIR/20200003/24.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
Include dependency graph for 24.c:
```

### Functions

- void [main](#) ()

#### 4.51.1 Function Documentation

##### 4.51.1.1 main()

```
void main (
    void )
```

## 4.52 STD\_DIR/20200006/24.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
Include dependency graph for 24.c:
```

### Functions

- void [main](#) ()

#### 4.52.1 Function Documentation

##### 4.52.1.1 main()

```
void main (
    void )
```

## 4.53 STD\_DIR/20200007/24.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
Include dependency graph for 24.c:
```

### Functions

- void [main](#) ()

#### 4.53.1 Function Documentation

##### 4.53.1.1 main()

```
void main (
    void )
```

## 4.54 STD\_DIR/20200008/24.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
Include dependency graph for 24.c:
```

### Functions

- void [main](#) ()

#### 4.54.1 Function Documentation

##### 4.54.1.1 main()

```
void main (
    void )
```

## 4.55 STD\_DIR/20200010/24.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
Include dependency graph for 24.c:
```

### Functions

- void [main](#) ()

#### 4.55.1 Function Documentation

##### 4.55.1.1 main()

```
void main (
    void )
```

## 4.56 20200002/25.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <signal.h>
Include dependency graph for 25.c:
```

### Functions

- int [main](#) (void)
- void [ssu\\_signal](#) (int signo)

### 4.56.1 Function Documentation

#### 4.56.1.1 main()

```
int main (
    void )
```

Here is the call graph for this function:

#### 4.56.1.2 ssu\_signal()

```
void ssu_signal (
    int signo )
```

Here is the caller graph for this function:

## 4.57 20200004/25.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <signal.h>
Include dependency graph for 25.c:
```

### Functions

- int [main](#) (void)
- void [ssu\\_signal](#) (int signo)



### 4.57.1 Function Documentation

#### 4.57.1.1 main()

```
int main (  
    void )
```

Here is the call graph for this function:

#### 4.57.1.2 ssu\_signal()

```
void ssu_signal (  
    int signo )
```

## 4.58 20200005/25.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <signal.h>  
Include dependency graph for 25.c:
```

### Functions

- int [main](#) (void)

### 4.58.1 Function Documentation

#### 4.58.1.1 main()

```
int main (  
    void )
```

Here is the call graph for this function:

## 4.59 20200009/25.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <signal.h>  
Include dependency graph for 25.c:
```

## Functions

- int [main](#) (void)
- void [ssu\\_signal](#) (int signo)

### 4.59.1 Function Documentation

#### 4.59.1.1 main()

```
int main (  
        void )
```

Here is the call graph for this function:

#### 4.59.1.2 ssu\_signal()

```
void ssu_signal (  
        int signo )
```

## 4.60 ANS\_DIR/25.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <signal.h>  
Include dependency graph for 25.c:
```

## Functions

- int [main](#) (void)
- void [ssu\\_signal](#) (int signo)

### 4.60.1 Function Documentation

#### 4.60.1.1 main()

```
int main (  
        void )
```

Here is the call graph for this function:

#### 4.60.1.2 ssu\_signal()

```
void ssu_signal (
    int signo )
```

## 4.61 STD\_DIR/20200001/25.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <signal.h>
Include dependency graph for 25.c:
```

### Functions

- int [main](#) (void)
- void [ssu\\_signal](#) (int signo)

#### 4.61.1 Function Documentation

##### 4.61.1.1 main()

```
int main (
    void )
```

Here is the call graph for this function:

##### 4.61.1.2 ssu\_signal()

```
void ssu_signal (
    int signo )
```

## 4.62 STD\_DIR/20200003/25.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <signal.h>
Include dependency graph for 25.c:
```

## Functions

- int [main](#) (void)
- void [ssu\\_signal](#) (int signo)

### 4.62.1 Function Documentation

#### 4.62.1.1 main()

```
int main (  
        void )
```

Here is the call graph for this function:

#### 4.62.1.2 ssu\_signal()

```
void ssu_signal (  
                int signo )
```

## 4.63 STD\_DIR/20200006/25.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <signal.h>  
Include dependency graph for 25.c:
```

## Functions

- int [main](#) (void)
- void [ssu\\_signal](#) (int signo)

### 4.63.1 Function Documentation

#### 4.63.1.1 main()

```
int main (  
        void )
```

Here is the call graph for this function:

#### 4.63.1.2 ssu\_signal()

```
void ssu_signal (
    int signo )
```

## 4.64 STD\_DIR/20200007/25.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <signal.h>
Include dependency graph for 25.c:
```

### Functions

- int [main](#) (void)
- void [ssu\\_signal](#) (int signo)

#### 4.64.1 Function Documentation

##### 4.64.1.1 main()

```
int main (
    void )
```

Here is the call graph for this function:

##### 4.64.1.2 ssu\_signal()

```
void ssu_signal (
    int signo )
```

## 4.65 STD\_DIR/20200008/25.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <signal.h>
Include dependency graph for 25.c:
```

## Functions

- int [main](#) (void)
- void [ssu\\_signal](#) (int signo)

### 4.65.1 Function Documentation

#### 4.65.1.1 main()

```
int main (  
        void )
```

Here is the call graph for this function:

#### 4.65.1.2 ssu\_signal()

```
void ssu_signal (  
        int signo )
```

## 4.66 STD\_DIR/20200010/25.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <signal.h>  
Include dependency graph for 25.c:
```

## Functions

- int [main](#) (void)
- void [ssu\\_signal](#) (int signo)

### 4.66.1 Function Documentation

#### 4.66.1.1 main()

```
int main (  
        void )
```

Here is the call graph for this function:

#### 4.66.1.2 ssu\_signal()

```
void ssu_signal (
    int signo )
```

## 4.67 20200002/26.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/syslog.h>
#include <sys/signal.h>
#include <sys/stat.h>
#include <fcntl.h>
```

Include dependency graph for 26.c:

### Functions

- int [main](#) (void)
- int [ssu\\_daemon\\_init](#) (void)

### 4.67.1 Function Documentation

#### 4.67.1.1 main()

```
int main (
    void )
```

Here is the call graph for this function:

#### 4.67.1.2 ssu\_daemon\_init()

```
int ssu_daemon_init (
    void )
```

Here is the caller graph for this function:

## 4.68 20200004/26.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/syslog.h>
#include <sys/signal.h>
#include <sys/stat.h>
#include <fcntl.h>
```

Include dependency graph for 26.c:

## Functions

- int [main](#) (void)
- int [ssu\\_daemon\\_init](#) (void)

### 4.68.1 Function Documentation

#### 4.68.1.1 main()

```
int main (  
        void )
```

Here is the call graph for this function:

#### 4.68.1.2 ssu\_daemon\_init()

```
int ssu_daemon_init (  
        void )
```

## 4.69 20200005/26.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <sys/syslog.h>  
#include <sys/signal.h>  
#include <sys/stat.h>  
#include <fcntl.h>
```

Include dependency graph for 26.c:

## Functions

- int [main](#) (void)
- int [ssu\\_daemon\\_init](#) (void)

### 4.69.1 Function Documentation

#### 4.69.1.1 main()

```
int main (  
        void )
```

Here is the call graph for this function:



#### 4.69.1.2 ssu\_daemon\_init()

```
int ssu_daemon_init (
    void )
```

## 4.70 20200009/26.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/syslog.h>
#include <sys/signal.h>
#include <sys/stat.h>
#include <fcntl.h>
```

Include dependency graph for 26.c:

### Functions

- int [main](#) (void)
- int [ssu\\_daemon\\_init](#) (void)

### 4.70.1 Function Documentation

#### 4.70.1.1 main()

```
int main (
    void )
```

Here is the call graph for this function:

#### 4.70.1.2 ssu\_daemon\_init()

```
int ssu_daemon_init (
    void )
```

## 4.71 ANS\_DIR/26.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/syslog.h>
#include <sys/signal.h>
#include <sys/stat.h>
#include <fcntl.h>
```

Include dependency graph for 26.c:

## Functions

- int [main](#) (void)
- int [ssu\\_daemon\\_init](#) (void)

### 4.71.1 Function Documentation

#### 4.71.1.1 main()

```
int main (  
        void )
```

Here is the call graph for this function:

#### 4.71.1.2 ssu\_daemon\_init()

```
int ssu_daemon_init (  
        void )
```

## 4.72 STD\_DIR/20200001/26.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <sys/syslog.h>  
#include <sys/signal.h>  
#include <sys/stat.h>  
#include <fcntl.h>
```

Include dependency graph for 26.c:

## Functions

- int [main](#) (void)
- int [ssu\\_daemon\\_init](#) (void)

### 4.72.1 Function Documentation

#### 4.72.1.1 main()

```
int main (  
        void )
```

Here is the call graph for this function:

#### 4.72.1.2 ssu\_daemon\_init()

```
int ssu_daemon_init (
    void )
```

## 4.73 STD\_DIR/20200003/26.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/syslog.h>
#include <sys/signal.h>
#include <sys/stat.h>
#include <fcntl.h>
```

Include dependency graph for 26.c:

### Functions

- int [main](#) (void)
- int [ssu\\_daemon\\_init](#) (void)

#### 4.73.1 Function Documentation

##### 4.73.1.1 main()

```
int main (
    void )
```

Here is the call graph for this function:

##### 4.73.1.2 ssu\_daemon\_init()

```
int ssu_daemon_init (
    void )
```

## 4.74 STD\_DIR/20200006/26.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/syslog.h>
#include <sys/signal.h>
#include <sys/stat.h>
#include <fcntl.h>
```

Include dependency graph for 26.c:

## Functions

- int [main](#) (void)
- int [ssu\\_daemon\\_init](#) (void)

### 4.74.1 Function Documentation

#### 4.74.1.1 main()

```
int main (  
        void )
```

Here is the call graph for this function:

#### 4.74.1.2 ssu\_daemon\_init()

```
int ssu_daemon_init (  
        void )
```

## 4.75 STD\_DIR/20200007/26.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <sys/syslog.h>  
#include <sys/signal.h>  
#include <sys/stat.h>  
#include <fcntl.h>
```

Include dependency graph for 26.c:

## Functions

- int [main](#) (void)
- int [ssu\\_daemon\\_init](#) (void)

### 4.75.1 Function Documentation

#### 4.75.1.1 main()

```
int main (  
        void )
```

Here is the call graph for this function:

#### 4.75.1.2 ssu\_daemon\_init()

```
int ssu_daemon_init (  
    void )
```

## 4.76 STD\_DIR/20200008/26.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <sys/syslog.h>  
#include <sys/signal.h>  
#include <sys/stat.h>  
#include <fcntl.h>
```

Include dependency graph for 26.c:

### Functions

- int [main](#) (void)
- int [ssu\\_daemon\\_init](#) (void)

### 4.76.1 Function Documentation

#### 4.76.1.1 main()

```
int main (  
    void )
```

Here is the call graph for this function:

#### 4.76.1.2 ssu\_daemon\_init()

```
int ssu_daemon_init (  
    void )
```

## 4.77 STD\_DIR/20200010/26.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <sys/syslog.h>  
#include <sys/signal.h>  
#include <sys/stat.h>  
#include <fcntl.h>
```

Include dependency graph for 26.c:

## Functions

- int [main](#) (void)
- int [ssu\\_daemon\\_init](#) (void)

### 4.77.1 Function Documentation

#### 4.77.1.1 main()

```
int main (  
        void )
```

Here is the call graph for this function:

#### 4.77.1.2 ssu\_daemon\_init()

```
int ssu_daemon_init (  
        void )
```

## 4.78 20200002/27.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <signal.h>  
Include dependency graph for 27.c:
```

## Functions

- int [main](#) (void)
- void [ssu\\_signal\\_handler](#) (int signo)

## Variables

- int [count](#) = 0

### 4.78.1 Function Documentation

#### 4.78.1.1 `main()`

```
int main (  
    void )
```

Here is the call graph for this function:

#### 4.78.1.2 `ssu_signal_handler()`

```
void ssu_signal_handler (  
    int signo )
```

Here is the caller graph for this function:

### 4.78.2 Variable Documentation

#### 4.78.2.1 `count`

```
int count = 0
```

## 4.79 20200004/27.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <signal.h>  
Include dependency graph for 27.c:
```

### Functions

- int [main](#) (void)
- void [ssu\\_signal\\_handler](#) (int signo)

### Variables

- int [count](#) = 0

### 4.79.1 Function Documentation

#### 4.79.1.1 main()

```
int main (  
    void )
```

Here is the call graph for this function:

#### 4.79.1.2 ssu\_signal\_handler()

```
void ssu_signal_handler (  
    int signo )
```

### 4.79.2 Variable Documentation

#### 4.79.2.1 count

```
int count = 0
```

## 4.80 20200005/27.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <signal.h>  
Include dependency graph for 27.c:
```

### Functions

- int [main](#) (void)
- void [ssu\\_signal\\_handler](#) (int signo)

### Variables

- int [count](#) = 0

#### 4.80.1 Function Documentation



#### 4.80.1.1 `main()`

```
int main (  
    void )
```

Here is the call graph for this function:

#### 4.80.1.2 `ssu_signal_handler()`

```
void ssu_signal_handler (  
    int signo )
```

### 4.80.2 Variable Documentation

#### 4.80.2.1 `count`

```
int count = 0
```

## 4.81 20200009/27.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <signal.h>  
Include dependency graph for 27.c:
```

### Functions

- int [main](#) (void)
- void [ssu\\_signal\\_handler](#) (int signo)

### Variables

- int [count](#) = 0

#### 4.81.1 Function Documentation

#### 4.81.1.1 main()

```
int main (  
    void )
```

Here is the call graph for this function:

#### 4.81.1.2 ssu\_signal\_handler()

```
void ssu_signal_handler (  
    int signo )
```

### 4.81.2 Variable Documentation

#### 4.81.2.1 count

```
int count = 0
```

## 4.82 ANS\_DIR/27.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <signal.h>  
Include dependency graph for 27.c:
```

### Functions

- int [main](#) (void)
- void [ssu\\_signal\\_handler](#) (int signo)

### Variables

- int [count](#) = 0

#### 4.82.1 Function Documentation

#### 4.82.1.1 main()

```
int main (  
    void )
```

Here is the call graph for this function:

#### 4.82.1.2 ssu\_signal\_handler()

```
void ssu_signal_handler (  
    int signo )
```

### 4.82.2 Variable Documentation

#### 4.82.2.1 count

```
int count = 0
```

## 4.83 STD\_DIR/20200001/27.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <signal.h>  
Include dependency graph for 27.c:
```

### Functions

- int [main](#) (void)
- void [ssu\\_signal\\_handler](#) (int signo)

### Variables

- int [count](#) = 0

### 4.83.1 Function Documentation

#### 4.83.1.1 `main()`

```
int main (  
    void )
```

Here is the call graph for this function:

#### 4.83.1.2 `ssu_signal_handler()`

```
void ssu_signal_handler (  
    int signo )
```

### 4.83.2 Variable Documentation

#### 4.83.2.1 `count`

```
int count = 0
```

## 4.84 `STD_DIR/20200003/27.c` File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <signal.h>  
Include dependency graph for 27.c:
```

### Functions

- int [main](#) (void)
- void [ssu\\_signal\\_handler](#) (int signo)

### Variables

- int [count](#) = 0

#### 4.84.1 Function Documentation

#### 4.84.1.1 main()

```
int main (  
    void )
```

Here is the call graph for this function:

#### 4.84.1.2 ssu\_signal\_handler()

```
void ssu_signal_handler (  
    int signo )
```

### 4.84.2 Variable Documentation

#### 4.84.2.1 count

```
int count = 0
```

## 4.85 STD\_DIR/20200006/27.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <signal.h>  
Include dependency graph for 27.c:
```

### Functions

- int [main](#) (void)
- void [ssu\\_signal\\_handler](#) (int signo)

### Variables

- int [count](#) = 0

### 4.85.1 Function Documentation

#### 4.85.1.1 main()

```
int main (  
    void )
```

Here is the call graph for this function:

#### 4.85.1.2 ssu\_signal\_handler()

```
void ssu_signal_handler (  
    int signo )
```

### 4.85.2 Variable Documentation

#### 4.85.2.1 count

```
int count = 0
```

## 4.86 STD\_DIR/20200007/27.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <signal.h>  
Include dependency graph for 27.c:
```

### Functions

- int [main](#) (void)
- void [ssu\\_signal\\_handler](#) (int signo)

### Variables

- int [count](#) = 0

#### 4.86.1 Function Documentation

#### 4.86.1.1 main()

```
int main (  
    void )
```

Here is the call graph for this function:

#### 4.86.1.2 ssu\_signal\_handler()

```
void ssu_signal_handler (  
    int signo )
```

### 4.86.2 Variable Documentation

#### 4.86.2.1 count

```
int count = 0
```

## 4.87 STD\_DIR/20200008/27.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <signal.h>  
Include dependency graph for 27.c:
```

### Functions

- int [main](#) (void)
- void [ssu\\_signal\\_handler](#) (int signo)

### Variables

- int [count](#) = 0

### 4.87.1 Function Documentation

#### 4.87.1.1 main()

```
int main (  
    void )
```

Here is the call graph for this function:

#### 4.87.1.2 ssu\_signal\_handler()

```
void ssu_signal_handler (  
    int signo )
```

### 4.87.2 Variable Documentation

#### 4.87.2.1 count

```
int count = 0
```

## 4.88 STD\_DIR/20200010/27.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <unistd.h>  
#include <signal.h>  
Include dependency graph for 27.c:
```

### Functions

- int [main](#) (void)
- void [ssu\\_signal\\_handler](#) (int signo)

### Variables

- int [count](#) = 0

#### 4.88.1 Function Documentation



#### 4.88.1.1 `main()`

```
int main (  
    void )
```

Here is the call graph for this function:

#### 4.88.1.2 `ssu_signal_handler()`

```
void ssu_signal_handler (  
    int signo )
```

### 4.88.2 Variable Documentation

#### 4.88.2.1 `count`

```
int count = 0
```

## 4.89 20200002/28.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <setjmp.h>  
Include dependency graph for 28.c:
```

### Functions

- int [main](#) (void)
- void [ssu\\_func](#) (int loc\_var, int loc\_volatile, int loc\_register)

### Variables

- int [count](#) = 0
- static jmp\_buf [glob\\_buffer](#)

### 4.89.1 Function Documentation

#### 4.89.1.1 main()

```
int main (
    void )
```

Here is the call graph for this function:

#### 4.89.1.2 ssu\_func()

```
void ssu_func (
    int loc_var,
    int loc_volatile,
    int loc_register )
```

Here is the caller graph for this function:

### 4.89.2 Variable Documentation

#### 4.89.2.1 count

```
int count = 0
```

#### 4.89.2.2 glob\_buffer

```
jmp_buf glob_buffer [static]
```

## 4.90 20200004/28.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <setjmp.h>
Include dependency graph for 28.c:
```

### Functions

- int [main](#) (void)
- void [ssu\\_func](#) (int loc\_var, int loc\_volatile, int loc\_register)

### Variables

- int [count](#) = 0
- static jmp\_buf [glob\\_buffer](#)

## 4.90.1 Function Documentation

### 4.90.1.1 `main()`

```
int main (  
    void )
```

Here is the call graph for this function:

### 4.90.1.2 `ssu_func()`

```
void ssu_func (  
    int loc_var,  
    int loc_volatile,  
    int loc_register )
```

Here is the call graph for this function: Here is the caller graph for this function:

## 4.90.2 Variable Documentation

### 4.90.2.1 `count`

```
int count = 0
```

### 4.90.2.2 `glob_buffer`

```
jmp_buf glob_buffer [static]
```

## 4.91 20200005/28.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <setjmp.h>  
Include dependency graph for 28.c:
```

## Functions

- int [main](#) (void)
- void [ssu\\_func](#) (int *loc\_var*, int *loc\_volatile*, int *loc\_register*)

## Variables

- int `count` = 0
- static jmp\_buf `glob_buffer`

## 4.91.1 Function Documentation

### 4.91.1.1 `main()`

```
int main (
    void )
```

Here is the call graph for this function:

### 4.91.1.2 `ssu_func()`

```
void ssu_func (
    int loc_var,
    int loc_volatile,
    int loc_register )
```

Here is the call graph for this function:

## 4.91.2 Variable Documentation

### 4.91.2.1 `count`

```
int count = 0
```

### 4.91.2.2 `glob_buffer`

```
jmp_buf glob_buffer [static]
```

## 4.92 20200009/28.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <setjmp.h>
Include dependency graph for 28.c:
```

## Functions

- int `main` (void)
- void `ssu_func` (int `loc_var`, int `loc_volatile`, int `loc_register`)

## Variables

- int `count` = 0
- static jmp\_buf `glob_buffer`

### 4.92.1 Function Documentation

#### 4.92.1.1 `main()`

```
int main (  
    void )
```

Here is the call graph for this function:

#### 4.92.1.2 `ssu_func()`

```
void ssu_func (  
    int loc_var,  
    int loc_volatile,  
    int loc_register )
```

Here is the call graph for this function:

### 4.92.2 Variable Documentation

#### 4.92.2.1 `count`

```
int count = 0
```

#### 4.92.2.2 `glob_buffer`

```
jmp_buf glob_buffer [static]
```

## 4.93 ANS\_DIR/28.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <setjmp.h>
Include dependency graph for 28.c:
```

### Functions

- int [main](#) (void)
- void [ssu\\_func](#) (int loc\_var, int loc\_volatile, int loc\_register)

### Variables

- int [count](#) = 0
- static jmp\_buf [glob\\_buffer](#)

### 4.93.1 Function Documentation

#### 4.93.1.1 main()

```
int main (
    void )
```

Here is the call graph for this function:

#### 4.93.1.2 ssu\_func()

```
void ssu_func (
    int loc_var,
    int loc_volatile,
    int loc_register )
```

Here is the call graph for this function:

### 4.93.2 Variable Documentation

#### 4.93.2.1 count

```
int count = 0
```

#### 4.93.2.2 glob\_buffer

```
jmp_buf glob_buffer [static]
```

## 4.94 STD\_DIR/20200001/28.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <setjmp.h>
Include dependency graph for 28.c:
```

### Functions

- int [main](#) (void)
- void [ssu\\_func](#) (int loc\_var, int loc\_volatile, int loc\_register)

### Variables

- int [count](#) = 0
- static jmp\_buf [glob\\_buffer](#)

### 4.94.1 Function Documentation

#### 4.94.1.1 main()

```
int main (
    void )
```

Here is the call graph for this function:

#### 4.94.1.2 ssu\_func()

```
void ssu_func (
    int loc_var,
    int loc_volatile,
    int loc_register )
```

Here is the call graph for this function:

### 4.94.2 Variable Documentation

#### 4.94.2.1 count

```
int count = 0
```

#### 4.94.2.2 glob\_buffer

```
jmp_buf glob_buffer [static]
```

### 4.95 STD\_DIR/20200003/28.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <setjmp.h>
Include dependency graph for 28.c:
```

#### Functions

- int [main](#) (void)
- void [ssu\\_func](#) (int loc\_var, int loc\_volatile, int loc\_register)

#### Variables

- int [count](#) = 0
- static jmp\_buf [glob\\_buffer](#)

#### 4.95.1 Function Documentation

##### 4.95.1.1 main()

```
int main (
    void )
```

Here is the call graph for this function:

##### 4.95.1.2 ssu\_func()

```
void ssu_func (
    int loc_var,
    int loc_volatile,
    int loc_register )
```

Here is the call graph for this function:



## 4.95.2 Variable Documentation

### 4.95.2.1 count

```
int count = 0
```

### 4.95.2.2 glob\_buffer

```
jmp_buf glob_buffer [static]
```

## 4.96 STD\_DIR/20200006/28.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <setjmp.h>
Include dependency graph for 28.c:
```

### Functions

- int [main](#) (void)
- void [ssu\\_func](#) (int loc\_var, int loc\_volatile, int loc\_register)

### Variables

- int [count](#) = 0
- static jmp\_buf [glob\\_buffer](#)

## 4.96.1 Function Documentation

### 4.96.1.1 main()

```
int main (
    void )
```

Here is the call graph for this function:

#### 4.96.1.2 ssu\_func()

```
void ssu_func (
    int loc_var,
    int loc_volatile,
    int loc_register )
```

Here is the call graph for this function:

### 4.96.2 Variable Documentation

#### 4.96.2.1 count

```
int count = 0
```

#### 4.96.2.2 glob\_buffer

```
jmp_buf glob_buffer [static]
```

## 4.97 STD\_DIR/20200007/28.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <setjmp.h>
Include dependency graph for 28.c:
```

### Functions

- int [main](#) (void)
- void [ssu\\_func](#) (int loc\_var, int loc\_volatile, int loc\_register)

### Variables

- int [count](#) = 0
- static jmp\_buf [glob\\_buffer](#)

#### 4.97.1 Function Documentation

#### 4.97.1.1 main()

```
int main (
    void )
```

Here is the call graph for this function:

#### 4.97.1.2 ssu\_func()

```
void ssu_func (
    int loc_var,
    int loc_volatile,
    int loc_register )
```

Here is the call graph for this function:

### 4.97.2 Variable Documentation

#### 4.97.2.1 count

```
int count = 0
```

#### 4.97.2.2 glob\_buffer

```
jmp_buf glob_buffer [static]
```

## 4.98 STD\_DIR/20200008/28.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <setjmp.h>
Include dependency graph for 28.c:
```

### Functions

- int [main](#) (void)
- void [ssu\\_func](#) (int loc\_var, int loc\_volatile, int loc\_register)

### Variables

- int [count](#) = 0
- static jmp\_buf [glob\\_buffer](#)

### 4.98.1 Function Documentation

#### 4.98.1.1 `main()`

```
int main (  
    void )
```

Here is the call graph for this function:

#### 4.98.1.2 `ssu_func()`

```
void ssu_func (  
    int loc_var,  
    int loc_volatile,  
    int loc_register )
```

Here is the call graph for this function:

### 4.98.2 Variable Documentation

#### 4.98.2.1 `count`

```
int count = 0
```

#### 4.98.2.2 `glob_buffer`

```
jmp_buf glob_buffer [static]
```

## 4.99 STD\_DIR/20200010/28.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <setjmp.h>  
Include dependency graph for 28.c:
```

### Functions

- int [main](#) (void)
- void [ssu\\_func](#) (int loc\_var, int loc\_volatile, int loc\_register)

## Variables

- int `count` = 0
- static jmp\_buf `glob_buffer`

## 4.99.1 Function Documentation

### 4.99.1.1 `main()`

```
int main (
    void )
```

Here is the call graph for this function:

### 4.99.1.2 `ssu_func()`

```
void ssu_func (
    int loc_var,
    int loc_volatile,
    int loc_register )
```

Here is the call graph for this function:

## 4.99.2 Variable Documentation

### 4.99.2.1 `count`

```
int count = 0
```

### 4.99.2.2 `glob_buffer`

```
jmp_buf glob_buffer [static]
```

## 4.100 20200002/29.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>
Include dependency graph for 29.c:
```

## Macros

- #define `VALUE_DONE` 10
- #define `VALUE_STOP1` 3
- #define `VALUE_STOP2` 6

## Functions

- int `main` (void)
- void \* `ssu_thread1` (void \*arg)
- void \* `ssu_thread2` (void \*arg)

## Variables

- pthread\_cond\_t `cond` = PTHREAD\_COND\_INITIALIZER
- int `glo_val` = 0
- pthread\_mutex\_t `lock` = PTHREAD\_MUTEX\_INITIALIZER

### 4.100.1 Macro Definition Documentation

#### 4.100.1.1 VALUE\_DONE

```
#define VALUE_DONE 10
```

#### 4.100.1.2 VALUE\_STOP1

```
#define VALUE_STOP1 3
```

#### 4.100.1.3 VALUE\_STOP2

```
#define VALUE_STOP2 6
```

### 4.100.2 Function Documentation

#### 4.100.2.1 main()

```
int main (
    void )
```

Here is the call graph for this function:

#### 4.100.2.2 ssu\_thread1()

```
void * ssu_thread1 (
    void * arg )
```

Here is the caller graph for this function:

#### 4.100.2.3 ssu\_thread2()

```
void * ssu_thread2 (
    void * arg )
```

Here is the caller graph for this function:

### 4.100.3 Variable Documentation

#### 4.100.3.1 cond

```
pthread_cond_t cond = PTHREAD_COND_INITIALIZER
```

#### 4.100.3.2 glo\_val

```
int glo_val = 0
```

#### 4.100.3.3 lock

```
pthread_mutex_t lock = PTHREAD_MUTEX_INITIALIZER
```

## 4.101 20200004/29.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>
Include dependency graph for 29.c:
```

## Macros

- `#define VALUE_DONE 10`
- `#define VALUE_STOP1 3`
- `#define VALUE_STOP2 6`

## Functions

- `int main (void)`
- `void * ssu_thread1 (void *arg)`
- `void * ssu_thread2 (void *arg)`

## Variables

- `pthread_cond_t cond = PTHREAD_COND_INITIALIZER`
- `int glo_val = 0`
- `pthread_mutex_t lock = PTHREAD_MUTEX_INITIALIZER`

### 4.101.1 Macro Definition Documentation

#### 4.101.1.1 VALUE\_DONE

```
#define VALUE_DONE 10
```

#### 4.101.1.2 VALUE\_STOP1

```
#define VALUE_STOP1 3
```

#### 4.101.1.3 VALUE\_STOP2

```
#define VALUE_STOP2 6
```

### 4.101.2 Function Documentation



#### 4.101.2.1 main()

```
int main (
    void )
```

Here is the call graph for this function:

#### 4.101.2.2 ssu\_thread1()

```
void* ssu_thread1 (
    void * arg )
```

#### 4.101.2.3 ssu\_thread2()

```
void* ssu_thread2 (
    void * arg )
```

### 4.101.3 Variable Documentation

#### 4.101.3.1 cond

```
pthread_cond_t cond = PTHREAD_COND_INITIALIZER
```

#### 4.101.3.2 glo\_val

```
int glo_val = 0
```

#### 4.101.3.3 lock

```
pthread_mutex_t lock = PTHREAD_MUTEX_INITIALIZER
```

## 4.102 20200005/29.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>
Include dependency graph for 29.c:
```

## Macros

- `#define VALUE_DONE 10`
- `#define VALUE_STOP1 3`
- `#define VALUE_STOP2 6`

## Functions

- `int main (void)`
- `void * ssu_thread1 (void *arg)`
- `void * ssu_thread2 (void *arg)`

## Variables

- `pthread_cond_t cond = PTHREAD_COND_INITIALIZER`
- `int glo_val = 0`
- `pthread_mutex_t lock = PTHREAD_MUTEX_INITIALIZER`

### 4.102.1 Macro Definition Documentation

#### 4.102.1.1 VALUE\_DONE

```
#define VALUE_DONE 10
```

#### 4.102.1.2 VALUE\_STOP1

```
#define VALUE_STOP1 3
```

#### 4.102.1.3 VALUE\_STOP2

```
#define VALUE_STOP2 6
```

### 4.102.2 Function Documentation

#### 4.102.2.1 main()

```
int main (
    void )
```

Here is the call graph for this function:

#### 4.102.2.2 ssu\_thread1()

```
void* ssu_thread1 (
    void * arg )
```

#### 4.102.2.3 ssu\_thread2()

```
void* ssu_thread2 (
    void * arg )
```

### 4.102.3 Variable Documentation

#### 4.102.3.1 cond

```
pthread_cond_t cond = PTHREAD_COND_INITIALIZER
```

#### 4.102.3.2 glo\_val

```
int glo_val = 0
```

#### 4.102.3.3 lock

```
pthread_mutex_t lock = PTHREAD_MUTEX_INITIALIZER
```

## 4.103 20200009/29.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>
Include dependency graph for 29.c:
```

## Macros

- `#define VALUE_DONE 10`
- `#define VALUE_STOP1 3`
- `#define VALUE_STOP2 6`

## Functions

- `int main (void)`
- `void * ssu_thread1 (void *arg)`
- `void * ssu_thread2 (void *arg)`

## Variables

- `pthread_cond_t cond = PTHREAD_COND_INITIALIZER`
- `int glo_val = 0`
- `pthread_mutex_t lock = PTHREAD_MUTEX_INITIALIZER`

### 4.103.1 Macro Definition Documentation

#### 4.103.1.1 VALUE\_DONE

```
#define VALUE_DONE 10
```

#### 4.103.1.2 VALUE\_STOP1

```
#define VALUE_STOP1 3
```

#### 4.103.1.3 VALUE\_STOP2

```
#define VALUE_STOP2 6
```

### 4.103.2 Function Documentation

#### 4.103.2.1 main()

```
int main (
    void )
```

Here is the call graph for this function:

#### 4.103.2.2 ssu\_thread1()

```
void* ssu_thread1 (
    void * arg )
```

#### 4.103.2.3 ssu\_thread2()

```
void* ssu_thread2 (
    void * arg )
```

### 4.103.3 Variable Documentation

#### 4.103.3.1 cond

```
pthread_cond_t cond = PTHREAD_COND_INITIALIZER
```

#### 4.103.3.2 glo\_val

```
int glo_val = 0
```

#### 4.103.3.3 lock

```
pthread_mutex_t lock = PTHREAD_MUTEX_INITIALIZER
```

## 4.104 ANS\_DIR/29.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>
Include dependency graph for 29.c:
```

## Macros

- `#define VALUE_DONE 10`
- `#define VALUE_STOP1 3`
- `#define VALUE_STOP2 6`

## Functions

- `int main (void)`
- `void * ssu_thread1 (void *arg)`
- `void * ssu_thread2 (void *arg)`

## Variables

- `pthread_cond_t cond = PTHREAD_COND_INITIALIZER`
- `int glo_val = 0`
- `pthread_mutex_t lock = PTHREAD_MUTEX_INITIALIZER`

### 4.104.1 Macro Definition Documentation

#### 4.104.1.1 VALUE\_DONE

```
#define VALUE_DONE 10
```

#### 4.104.1.2 VALUE\_STOP1

```
#define VALUE_STOP1 3
```

#### 4.104.1.3 VALUE\_STOP2

```
#define VALUE_STOP2 6
```

### 4.104.2 Function Documentation

#### 4.104.2.1 main()

```
int main (  
    void )
```

Here is the call graph for this function:

#### 4.104.2.2 ssu\_thread1()

```
void* ssu_thread1 (  
    void * arg )
```

#### 4.104.2.3 ssu\_thread2()

```
void* ssu_thread2 (  
    void * arg )
```

### 4.104.3 Variable Documentation

#### 4.104.3.1 cond

```
pthread_cond_t cond = PTHREAD_COND_INITIALIZER
```

#### 4.104.3.2 glo\_val

```
int glo_val = 0
```

#### 4.104.3.3 lock

```
pthread_mutex_t lock = PTHREAD_MUTEX_INITIALIZER
```

## 4.105 STD\_DIR/20200001/29.c File Reference

```
#include <stdio.h>  
#include <stdlib.h>  
#include <pthread.h>  
Include dependency graph for 29.c:
```

## Macros

- `#define VALUE_DONE 10`
- `#define VALUE_STOP1 3`
- `#define VALUE_STOP2 6`

## Functions

- `int main (void)`
- `void * ssu_thread1 (void *arg)`
- `void * ssu_thread2 (void *arg)`

## Variables

- `pthread_cond_t cond = PTHREAD_COND_INITIALIZER`
- `int glo_val = 0`
- `pthread_mutex_t lock = PTHREAD_MUTEX_INITIALIZER`

### 4.105.1 Macro Definition Documentation

#### 4.105.1.1 VALUE\_DONE

```
#define VALUE_DONE 10
```

#### 4.105.1.2 VALUE\_STOP1

```
#define VALUE_STOP1 3
```

#### 4.105.1.3 VALUE\_STOP2

```
#define VALUE_STOP2 6
```

### 4.105.2 Function Documentation



#### 4.105.2.1 main()

```
int main (
    void )
```

Here is the call graph for this function:

#### 4.105.2.2 ssu\_thread1()

```
void* ssu_thread1 (
    void * arg )
```

#### 4.105.2.3 ssu\_thread2()

```
void* ssu_thread2 (
    void * arg )
```

### 4.105.3 Variable Documentation

#### 4.105.3.1 cond

```
pthread_cond_t cond = PTHREAD_COND_INITIALIZER
```

#### 4.105.3.2 glo\_val

```
int glo_val = 0
```

#### 4.105.3.3 lock

```
pthread_mutex_t lock = PTHREAD_MUTEX_INITIALIZER
```

## 4.106 STD\_DIR/20200003/29.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>
Include dependency graph for 29.c:
```

## Macros

- `#define VALUE_DONE 10`
- `#define VALUE_STOP1 3`
- `#define VALUE_STOP2 6`

## Functions

- `int main (void)`
- `void * ssu_thread1 (void *arg)`
- `void * ssu_thread2 (void *arg)`

## Variables

- `pthread_cond_t cond = PTHREAD_COND_INITIALIZER`
- `int glo_val = 0`
- `pthread_mutex_t lock = PTHREAD_MUTEX_INITIALIZER`

### 4.106.1 Macro Definition Documentation

#### 4.106.1.1 VALUE\_DONE

```
#define VALUE_DONE 10
```

#### 4.106.1.2 VALUE\_STOP1

```
#define VALUE_STOP1 3
```

#### 4.106.1.3 VALUE\_STOP2

```
#define VALUE_STOP2 6
```

### 4.106.2 Function Documentation

#### 4.106.2.1 main()

```
int main (
    void )
```

Here is the call graph for this function:

#### 4.106.2.2 ssu\_thread1()

```
void* ssu_thread1 (
    void * arg )
```

#### 4.106.2.3 ssu\_thread2()

```
void* ssu_thread2 (
    void * arg )
```

### 4.106.3 Variable Documentation

#### 4.106.3.1 cond

```
pthread_cond_t cond = PTHREAD_COND_INITIALIZER
```

#### 4.106.3.2 glo\_val

```
int glo_val = 0
```

#### 4.106.3.3 lock

```
pthread_mutex_t lock = PTHREAD_MUTEX_INITIALIZER
```

## 4.107 STD\_DIR/20200006/29.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>
Include dependency graph for 29.c:
```

## Macros

- `#define VALUE_DONE 10`
- `#define VALUE_STOP1 3`
- `#define VALUE_STOP2 6`

## Functions

- `int main (void)`
- `void * ssu_thread1 (void *arg)`
- `void * ssu_thread2 (void *arg)`

## Variables

- `pthread_cond_t cond = PTHREAD_COND_INITIALIZER`
- `int glo_val = 0`
- `pthread_mutex_t lock = PTHREAD_MUTEX_INITIALIZER`

### 4.107.1 Macro Definition Documentation

#### 4.107.1.1 VALUE\_DONE

```
#define VALUE_DONE 10
```

#### 4.107.1.2 VALUE\_STOP1

```
#define VALUE_STOP1 3
```

#### 4.107.1.3 VALUE\_STOP2

```
#define VALUE_STOP2 6
```

### 4.107.2 Function Documentation

#### 4.107.2.1 main()

```
int main (
    void )
```

Here is the call graph for this function:

#### 4.107.2.2 ssu\_thread1()

```
void* ssu_thread1 (
    void * arg )
```

#### 4.107.2.3 ssu\_thread2()

```
void* ssu_thread2 (
    void * arg )
```

### 4.107.3 Variable Documentation

#### 4.107.3.1 cond

```
pthread_cond_t cond = PTHREAD_COND_INITIALIZER
```

#### 4.107.3.2 glo\_val

```
int glo_val = 0
```

#### 4.107.3.3 lock

```
pthread_mutex_t lock = PTHREAD_MUTEX_INITIALIZER
```

## 4.108 STD\_DIR/20200007/29.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>
Include dependency graph for 29.c:
```

## Macros

- `#define` [VALUE\\_DONE](#) 10
- `#define` [VALUE\\_STOP1](#) 3
- `#define` [VALUE\\_STOP2](#) 6

## Functions

- `int` [main](#) (void)
- `void *` [ssu\\_thread1](#) (void \*arg)
- `void *` [ssu\\_thread2](#) (void \*arg)

## Variables

- `pthread_cond_t` [cond](#) = PTHREAD\_COND\_INITIALIZER
- `int` [glo\\_val](#) = 0
- `pthread_mutex_t` [lock](#) = PTHREAD\_MUTEX\_INITIALIZER

### 4.108.1 Macro Definition Documentation

#### 4.108.1.1 VALUE\_DONE

```
#define VALUE_DONE 10
```

#### 4.108.1.2 VALUE\_STOP1

```
#define VALUE_STOP1 3
```

#### 4.108.1.3 VALUE\_STOP2

```
#define VALUE_STOP2 6
```

### 4.108.2 Function Documentation

#### 4.108.2.1 main()

```
int main (
    void )
```

Here is the call graph for this function:

#### 4.108.2.2 ssu\_thread1()

```
void* ssu_thread1 (
    void * arg )
```

#### 4.108.2.3 ssu\_thread2()

```
void* ssu_thread2 (
    void * arg )
```

### 4.108.3 Variable Documentation

#### 4.108.3.1 cond

```
pthread_cond_t cond = PTHREAD_COND_INITIALIZER
```

#### 4.108.3.2 glo\_val

```
int glo_val = 0
```

#### 4.108.3.3 lock

```
pthread_mutex_t lock = PTHREAD_MUTEX_INITIALIZER
```

## 4.109 STD\_DIR/20200008/29.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>
Include dependency graph for 29.c:
```

## Macros

- `#define VALUE_DONE 10`
- `#define VALUE_STOP1 3`
- `#define VALUE_STOP2 6`

## Functions

- `int main (void)`
- `void * ssu_thread1 (void *arg)`
- `void * ssu_thread2 (void *arg)`

## Variables

- `pthread_cond_t cond = PTHREAD_COND_INITIALIZER`
- `int glo_val = 0`
- `pthread_mutex_t lock = PTHREAD_MUTEX_INITIALIZER`

### 4.109.1 Macro Definition Documentation

#### 4.109.1.1 VALUE\_DONE

```
#define VALUE_DONE 10
```

#### 4.109.1.2 VALUE\_STOP1

```
#define VALUE_STOP1 3
```

#### 4.109.1.3 VALUE\_STOP2

```
#define VALUE_STOP2 6
```

### 4.109.2 Function Documentation



#### 4.109.2.1 main()

```
int main (
    void )
```

Here is the call graph for this function:

#### 4.109.2.2 ssu\_thread1()

```
void* ssu_thread1 (
    void * arg )
```

#### 4.109.2.3 ssu\_thread2()

```
void* ssu_thread2 (
    void * arg )
```

### 4.109.3 Variable Documentation

#### 4.109.3.1 cond

```
pthread_cond_t cond = PTHREAD_COND_INITIALIZER
```

#### 4.109.3.2 glo\_val

```
int glo_val = 0
```

#### 4.109.3.3 lock

```
pthread_mutex_t lock = PTHREAD_MUTEX_INITIALIZER
```

## 4.110 STD\_DIR/20200010/29.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>
Include dependency graph for 29.c:
```

## Macros

- `#define VALUE_DONE 10`
- `#define VALUE_STOP1 3`
- `#define VALUE_STOP2 6`

## Functions

- `int main (void)`
- `void * ssu_thread1 (void *arg)`
- `void * ssu_thread2 (void *arg)`

## Variables

- `pthread_cond_t cond = PTHREAD_COND_INITIALIZER`
- `int glo_val = 0`
- `pthread_mutex_t lock = PTHREAD_MUTEX_INITIALIZER`

### 4.110.1 Macro Definition Documentation

#### 4.110.1.1 VALUE\_DONE

```
#define VALUE_DONE 10
```

#### 4.110.1.2 VALUE\_STOP1

```
#define VALUE_STOP1 3
```

#### 4.110.1.3 VALUE\_STOP2

```
#define VALUE_STOP2 6
```

### 4.110.2 Function Documentation

#### 4.110.2.1 main()

```
int main (
    void )
```

Here is the call graph for this function:

#### 4.110.2.2 ssu\_thread1()

```
void* ssu_thread1 (
    void * arg )
```

#### 4.110.2.3 ssu\_thread2()

```
void* ssu_thread2 (
    void * arg )
```

### 4.110.3 Variable Documentation

#### 4.110.3.1 cond

```
pthread_cond_t cond = PTHREAD_COND_INITIALIZER
```

#### 4.110.3.2 glo\_val

```
int glo_val = 0
```

#### 4.110.3.3 lock

```
pthread_mutex_t lock = PTHREAD_MUTEX_INITIALIZER
```

## 4.111 blank.c File Reference

```
#include <stdio.h>
#include <string.h>
#include <unistd.h>
#include <stdlib.h>
#include <ctype.h>
#include "blank.h"
```

Include dependency graph for blank.c:

## Functions

- int [all\\_character](#) (char \*str)
- int [all\\_star](#) (char \*str)
- node \* [change\\_sibling](#) (node \*parent)
- int [check\\_brackets](#) (char \*str)
- void [clear\\_tokens](#) (char tokens[TOKEN\_CNT][MINLEN])
- void [compare\\_tree](#) (node \*root1, node \*root2, int \*result)
- node \* [create\\_node](#) (char \*name, int parentheses)
- int [find\\_typeSpecifier](#) (char tokens[TOKEN\_CNT][MINLEN])
- int [find\\_typeSpecifier2](#) (char tokens[TOKEN\_CNT][MINLEN])
- void [free\\_node](#) (node \*cur)
- node \* [get\\_high\\_precedence\\_node](#) (node \*cur, node \*new)
- node \* [get\\_last\\_child](#) (node \*cur)
- node \* [get\\_most\\_high\\_precedence\\_node](#) (node \*cur, node \*new)
- node \* [get\\_operator](#) (node \*cur)
- int [get\\_precedence](#) (char \*op)
- node \* [get\\_root](#) (node \*cur)
- int [get\\_sibling\\_cnt](#) (node \*cur)
- int [get\\_token\\_cnt](#) (char tokens[TOKEN\_CNT][MINLEN])
- node \* [insert\\_node](#) (node \*old, node \*new)
- int [is\\_character](#) (char c)
- int [is\\_operator](#) (char \*op)
- int [is\\_typeStatement](#) (char \*str)
- char \* [ltrim](#) (char \*\_str)
- int [make\\_tokens](#) (char \*str, char tokens[TOKEN\_CNT][MINLEN])
- node \* [make\\_tree](#) (node \*root, char(\*tokens)[MINLEN], int \*idx, int parentheses)
- char \* [remove\\_extraspace](#) (char \*str)
- void [remove\\_space](#) (char \*str)
- int [reset\\_tokens](#) (int start, char tokens[TOKEN\_CNT][MINLEN])
- char \* [rtrim](#) (char \*\_str)

## Variables

- char [datatype](#) [DATATYPE\_SIZE][MINLEN]
- [operator\\_precedence operators](#) [OPERATOR\_CNT]

### 4.111.1 Function Documentation

#### 4.111.1.1 [all\\_character\(\)](#)

```
int all_character (
    char * str )
```

Here is the call graph for this function:

#### 4.111.1.2 all\_star()

```
int all_star (
    char * str )
```

Here is the caller graph for this function:

#### 4.111.1.3 change\_sibling()

```
node* change_sibling (
    node * parent )
```

Here is the caller graph for this function:

#### 4.111.1.4 check\_brackets()

```
int check_brackets (
    char * str )
```

Here is the caller graph for this function:

#### 4.111.1.5 clear\_tokens()

```
void clear_tokens (
    char tokens[TOKEN_CNT][MINLEN] )
```

Here is the caller graph for this function:

#### 4.111.1.6 compare\_tree()

```
void compare_tree (
    node * root1,
    node * root2,
    int * result )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.111.1.7 create\_node()

```
node* create_node (
    char * name,
    int parentheses )
```

Here is the caller graph for this function:

#### 4.111.1.8 find\_typeSpecifier()

```
int find_typeSpecifier (
    char tokens[TOKEN_CNT][MINLEN] )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.111.1.9 find\_typeSpecifier2()

```
int find_typeSpecifier2 (
    char tokens[TOKEN_CNT][MINLEN] )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.111.1.10 free\_node()

```
void free_node (
    node * cur )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.111.1.11 get\_high\_precedence\_node()

```
node* get_high_precedence_node (
    node * cur,
    node * new )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.111.1.12 get\_last\_child()

```
node* get_last_child (
    node * cur )
```

Here is the caller graph for this function:

#### 4.111.1.13 get\_most\_high\_precedence\_node()

```
node* get_most_high_precedence_node (
    node * cur,
    node * new )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.111.1.14 get\_operator()

```
node* get_operator (
    node * cur )
```

Here is the caller graph for this function:

#### 4.111.1.15 get\_precedence()

```
int get_precedence (
    char * op )
```

Here is the caller graph for this function:

#### 4.111.1.16 get\_root()

```
node* get_root (
    node * cur )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.111.1.17 get\_sibling\_cnt()

```
int get_sibling_cnt (
    node * cur )
```

Here is the caller graph for this function:

#### 4.111.1.18 get\_token\_cnt()

```
int get_token_cnt (
    char tokens[TOKEN_CNT][MINLEN] )
```

#### 4.111.1.19 insert\_node()

```
node* insert_node (
    node * old,
    node * new )
```

Here is the caller graph for this function:

#### 4.111.1.20 is\_character()

```
int is_character (
    char c )
```

Here is the caller graph for this function:

#### 4.111.1.21 is\_operator()

```
int is_operator (
    char * op )
```

Here is the caller graph for this function:

#### 4.111.1.22 is\_typeStatement()

```
int is_typeStatement (
    char * str )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.111.1.23 ltrim()

```
char* ltrim (
    char * _str )
```

Here is the caller graph for this function:

#### 4.111.1.24 make\_tokens()

```
int make_tokens (
    char * str,
    char tokens[TOKEN_CNT][MINLEN] )
```

Here is the caller graph for this function:

#### 4.111.1.25 make\_tree()

```
node* make_tree (
    node * root,
    char(*) tokens[MINLEN],
    int * idx,
    int parentheses )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.111.1.26 remove\_extraspace()

```
char* remove_extraspace (
    char * str )
```

Here is the caller graph for this function:

#### 4.111.1.27 remove\_space()

```
void remove_space (
    char * str )
```

Here is the caller graph for this function:



## 4.111.1.28 reset\_tokens()

```
int reset_tokens (
    int start,
    char tokens[TOKEN_CNT][MINLEN] )
```

Here is the call graph for this function: Here is the caller graph for this function:

## 4.111.1.29 rtrim()

```
char* rtrim (
    char * _str )
```

Here is the caller graph for this function:

## 4.111.2 Variable Documentation

## 4.111.2.1 datatype

```
char datatype[DATATYPE_SIZE][MINLEN]
```

**Initial value:**

```
= { "int", "char", "double", "float", "long"
    , "short", "ushort", "FILE", "DIR", "pid"
    , "key_t", "ssize_t", "mode_t", "ino_t", "dev_t"
    , "nlink_t", "uid_t", "gid_t", "time_t", "blksize_t"
    , "blkcnt_t", "pid_t", "pthread_mutex_t", "pthread_cond_t", "pthread_t"
    , "void", "size_t", "unsigned", "sigset_t", "sigjmp_buf"
    , "rlim_t", "jmp_buf", "sig_atomic_t", "clock_t", "struct" }
```

## 4.111.2.2 operators

```
operator_precedence operators[OPERATOR_CNT]
```

**Initial value:**

```
= {
    { "(", 0 }, { ")", 0 }
    , { ">", 1 }
    , { "*", 4 } , { "/", 3 } , { "%", 2 }
    , { "+", 6 } , { "-", 5 }
    , { "<", 7 } , { "<=", 7 } , { ">", 7 } , { ">=", 7 }
    , { "=", 8 } , { "!=", 8 }
    , { "&", 9 }
    , { "^", 10 }
    , { "|", 11 }
    , { "&&", 12 }
    , { "||", 13 }
    , { "=", 14 } , { "+=", 14 } , { "-=", 14 } , { "&=", 14 } , { "|=", 14 }
}
```

## 4.112 blank.h File Reference

This graph shows which files directly or indirectly include this file:

### Classes

- struct [node](#)
- struct [operator\\_precedence](#)

### Macros

- #define [BUFLEN](#) 1024
- #define [DATATYPE\\_SIZE](#) 35
- #define [false](#) 0
- #define [MINLEN](#) 64
- #define [OPERATOR\\_CNT](#) 24
- #define [TOKEN\\_CNT](#) 50
- #define [true](#) 1

### Typedefs

- typedef struct [node](#) [node](#)
- typedef struct [operator\\_precedence](#) [operator\\_precedence](#)

### Functions

- int [all\\_character](#) (char \*str)
- int [all\\_star](#) (char \*str)
- [node](#) \* [change\\_sibling](#) ([node](#) \*parent)
- int [check\\_brackets](#) (char \*str)
- void [clear\\_tokens](#) (char tokens[[TOKEN\\_CNT](#)][[MINLEN](#)])
- void [compare\\_tree](#) ([node](#) \*root1, [node](#) \*root2, int \*result)
- [node](#) \* [create\\_node](#) (char \*name, int parentheses)
- int [find\\_typeSpecifier](#) (char tokens[[TOKEN\\_CNT](#)][[MINLEN](#)])
- int [find\\_typeSpecifier2](#) (char tokens[[TOKEN\\_CNT](#)][[MINLEN](#)])
- void [free\\_node](#) ([node](#) \*cur)
- [node](#) \* [get\\_high\\_precedence\\_node](#) ([node](#) \*cur, [node](#) \*new)
- [node](#) \* [get\\_last\\_child](#) ([node](#) \*cur)
- [node](#) \* [get\\_most\\_high\\_precedence\\_node](#) ([node](#) \*cur, [node](#) \*new)
- [node](#) \* [get\\_operator](#) ([node](#) \*cur)
- int [get\\_precedence](#) (char \*op)
- [node](#) \* [get\\_root](#) ([node](#) \*cur)
- int [get\\_sibling\\_cnt](#) ([node](#) \*cur)
- int [get\\_token\\_cnt](#) (char tokens[[TOKEN\\_CNT](#)][[MINLEN](#)])
- [node](#) \* [insert\\_node](#) ([node](#) \*old, [node](#) \*new)
- int [is\\_character](#) (char c)
- int [is\\_operator](#) (char \*op)
- int [is\\_typeStatement](#) (char \*str)
- char \* [ltrim](#) (char \*\_str)
- int [make\\_tokens](#) (char \*str, char tokens[[TOKEN\\_CNT](#)][[MINLEN](#)])
- [node](#) \* [make\\_tree](#) ([node](#) \*root, char(\*tokens)[[MINLEN](#)], int \*idx, int parentheses)
- void [print](#) ([node](#) \*cur)
- char \* [remove\\_extraspace](#) (char \*str)
- void [remove\\_space](#) (char \*str)
- int [reset\\_tokens](#) (int start, char tokens[[TOKEN\\_CNT](#)][[MINLEN](#)])
- char \* [rtrim](#) (char \*\_str)

### 4.112.1 Macro Definition Documentation

#### 4.112.1.1 BUFLLEN

```
#define BUFLLEN 1024
```

#### 4.112.1.2 DATATYPE\_SIZE

```
#define DATATYPE_SIZE 35
```

#### 4.112.1.3 false

```
#define false 0
```

#### 4.112.1.4 MINLEN

```
#define MINLEN 64
```

#### 4.112.1.5 OPERATOR\_CNT

```
#define OPERATOR_CNT 24
```

#### 4.112.1.6 TOKEN\_CNT

```
#define TOKEN_CNT 50
```

#### 4.112.1.7 true

```
#define true 1
```

## 4.112.2 Typedef Documentation

### 4.112.2.1 node

```
typedef struct node node
```

### 4.112.2.2 operator\_precedence

```
typedef struct operator_precedence operator_precedence
```

## 4.112.3 Function Documentation

### 4.112.3.1 all\_character()

```
int all_character (
    char * str )
```

Here is the call graph for this function:

### 4.112.3.2 all\_star()

```
int all_star (
    char * str )
```

Here is the caller graph for this function:

### 4.112.3.3 change\_sibling()

```
node* change_sibling (
    node * parent )
```

Here is the caller graph for this function:

### 4.112.3.4 check\_brackets()

```
int check_brackets (
    char * str )
```

Here is the caller graph for this function:

#### 4.112.3.5 clear\_tokens()

```
void clear_tokens (
    char tokens[TOKEN_CNT][MINLEN] )
```

Here is the caller graph for this function:

#### 4.112.3.6 compare\_tree()

```
void compare_tree (
    node * root1,
    node * root2,
    int * result )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.112.3.7 create\_node()

```
node* create_node (
    char * name,
    int parentheses )
```

Here is the caller graph for this function:

#### 4.112.3.8 find\_typeSpecifier()

```
int find_typeSpecifier (
    char tokens[TOKEN_CNT][MINLEN] )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.112.3.9 find\_typeSpecifier2()

```
int find_typeSpecifier2 (
    char tokens[TOKEN_CNT][MINLEN] )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.112.3.10 free\_node()

```
void free_node (
    node * cur )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.112.3.11 `get_high_precedence_node()`

```
node* get_high_precedence_node (
    node * cur,
    node * new )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.112.3.12 `get_last_child()`

```
node* get_last_child (
    node * cur )
```

Here is the caller graph for this function:

#### 4.112.3.13 `get_most_high_precedence_node()`

```
node* get_most_high_precedence_node (
    node * cur,
    node * new )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.112.3.14 `get_operator()`

```
node* get_operator (
    node * cur )
```

Here is the caller graph for this function:

#### 4.112.3.15 `get_precedence()`

```
int get_precedence (
    char * op )
```

Here is the caller graph for this function:

#### 4.112.3.16 `get_root()`

```
node* get_root (
    node * cur )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.112.3.17 `get_sibling_cnt()`

```
int get_sibling_cnt (
    node * cur )
```

Here is the caller graph for this function:

#### 4.112.3.18 get\_token\_cnt()

```
int get_token_cnt (
    char tokens[TOKEN_CNT][MINLEN] )
```

#### 4.112.3.19 insert\_node()

```
node* insert_node (
    node * old,
    node * new )
```

Here is the caller graph for this function:

#### 4.112.3.20 is\_character()

```
int is_character (
    char c )
```

Here is the caller graph for this function:

#### 4.112.3.21 is\_operator()

```
int is_operator (
    char * op )
```

Here is the caller graph for this function:

#### 4.112.3.22 is\_typeStatement()

```
int is_typeStatement (
    char * str )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.112.3.23 ltrim()

```
char* ltrim (
    char * _str )
```

Here is the caller graph for this function:

#### 4.112.3.24 make\_tokens()

```
int make_tokens (
    char * str,
    char tokens[TOKEN_CNT][MINLEN] )
```

Here is the caller graph for this function:

**4.112.3.25 make\_tree()**

```
node* make_tree (
    node * root,
    char(*) tokens[MINLEN],
    int * idx,
    int parentheses )
```

Here is the call graph for this function: Here is the caller graph for this function:

**4.112.3.26 print()**

```
void print (
    node * cur )
```

**4.112.3.27 remove\_extraspace()**

```
char* remove_extraspace (
    char * str )
```

Here is the caller graph for this function:

**4.112.3.28 remove\_space()**

```
void remove_space (
    char * str )
```

Here is the caller graph for this function:

**4.112.3.29 reset\_tokens()**

```
int reset_tokens (
    int start,
    char tokens[TOKEN_CNT][MINLEN] )
```

Here is the call graph for this function: Here is the caller graph for this function:

**4.112.3.30 rtrim()**

```
char* rtrim (
    char * _str )
```

Here is the caller graph for this function:



## 4.113 main.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/time.h>
#include "ssu_score.h"
Include dependency graph for main.c:
```

### Macros

- #define [SECOND\\_TO\\_MICRO](#) 1000000

### Functions

- int [main](#) (int argc, char \*argv[])
- void [ssu\\_runtime](#) (struct timeval \*begin\_t, struct timeval \*end\_t)

#### 4.113.1 Macro Definition Documentation

##### 4.113.1.1 SECOND\_TO\_MICRO

```
#define SECOND_TO_MICRO 1000000
```

#### 4.113.2 Function Documentation

##### 4.113.2.1 main()

```
int main (
    int argc,
    char * argv[] )
```

Here is the call graph for this function:

##### 4.113.2.2 ssu\_runtime()

```
void ssu_runtime (
    struct timeval * begin_t,
    struct timeval * end_t )
```

Here is the caller graph for this function:

## 4.114 ssu\_score.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <signal.h>
#include <string.h>
#include <sys/types.h>
#include <dirent.h>
#include <unistd.h>
#include <fcntl.h>
#include <sys/stat.h>
#include "ssu_score.h"
#include "blank.h"
```

### Functions

- double [check\\_error\\_warning](#) (char \*filename)
- int [check\\_option](#) (int argc, char \*argv[])
- int [compare\\_resultfile](#) (char \*file1, char \*file2)
- double [compile\\_program](#) (char \*id, char \*filename)
- void [do\\_iOption](#) (int argc, char \*argv[], int optind)
- void [do\\_mOption](#) (char \*path)
- int [execute\\_program](#) (char \*id, char \*filename)
- char \* [get\\_answer](#) (int fd, char \*result)
- int [get\\_create\\_type](#) ()
- int [get\\_file\\_type](#) (char \*filename)
- void [get\\_qname\\_number](#) (char \*qname, int \*num1, int \*num2)
- pid\_t [inBackground](#) (char \*name)
- int [is\\_thread](#) (char \*qname)
- void [make\\_scoreTable](#) (char \*ansDir)
- void [print\\_usage](#) ()
- void [read\\_scoreTable](#) (char \*path)
- void [redirection](#) (char \*command, int new, int old)
- void [rmdirs](#) (const char \*path)
- int [score\\_blank](#) (char \*id, char \*filename)
- double [score\\_program](#) (char \*id, char \*filename)
- double [score\\_student](#) (int fd, char \*id)
- void [score\\_students](#) ()
- void [set\\_idTable](#) (char \*stuDir)
- void [set\\_scoreTable](#) (char \*ansDir)
- void [sort\\_idTable](#) (int size)
- void [sort\\_scoreTable](#) (int size)
- void [ssu\\_score](#) (int argc, char \*argv[])
- void [to\\_lower\\_case](#) (char \*c)
- void [write\\_first\\_row](#) (int fd)
- void [write\\_scoreTable](#) (char \*filename)

## Variables

- char `ansDir` [BUFLEN]
- int `eOption` = false
- `-i`
- char `errorDir` [BUFLEN]
- char `id_table` [SNUM][10]
- int `iOption` = false
- int `mOption` = false
- char `printId` [ARGNUM][FILELEN]
- struct `ssu_scoreTable` `score_table` [QNUM]
- char `stuDir` [BUFLEN]
- char `threadFiles` [ARGNUM][FILELEN]
- int `tOption` = false

## 4.114.1 Function Documentation

### 4.114.1.1 `check_error_warning()`

```
double check_error_warning (
    char * filename )
```

Here is the caller graph for this function:

### 4.114.1.2 `check_option()`

```
int check_option (
    int argc,
    char * argv[] )
```

Here is the call graph for this function: Here is the caller graph for this function:

### 4.114.1.3 `compare_resultfile()`

```
int compare_resultfile (
    char * file1,
    char * file2 )
```

Here is the call graph for this function: Here is the caller graph for this function:

### 4.114.1.4 `compile_program()`

```
double compile_program (
    char * id,
    char * filename )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.114.1.5 do\_iOption()

```
void do_iOption (
    int argc,
    char * argv[],
    int optind )
```

Here is the caller graph for this function:

#### 4.114.1.6 do\_mOption()

```
void do_mOption (
    char * path )
```

Here is the caller graph for this function:

#### 4.114.1.7 execute\_program()

```
int execute_program (
    char * id,
    char * filename )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.114.1.8 get\_answer()

```
char* get_answer (
    int fd,
    char * result )
```

Here is the caller graph for this function:

#### 4.114.1.9 get\_create\_type()

```
int get_create_type ( )
```

Here is the caller graph for this function:

#### 4.114.1.10 get\_file\_type()

```
int get_file_type (
    char * filename )
```

Here is the caller graph for this function:

**4.114.1.11 get\_qname\_number()**

```
void get_qname_number (
    char * qname,
    int * num1,
    int * num2 )
```

Here is the caller graph for this function:

**4.114.1.12 inBackground()**

```
pid_t inBackground (
    char * name )
```

Here is the call graph for this function: Here is the caller graph for this function:

**4.114.1.13 is\_thread()**

```
int is_thread (
    char * qname )
```

Here is the caller graph for this function:

**4.114.1.14 make\_scoreTable()**

```
void make_scoreTable (
    char * ansDir )
```

Here is the call graph for this function: Here is the caller graph for this function:

**4.114.1.15 print\_usage()**

```
void print_usage ( )
```

Here is the caller graph for this function:

**4.114.1.16 read\_scoreTable()**

```
void read_scoreTable (
    char * path )
```

Here is the caller graph for this function:

**4.114.1.17 redirection()**

```
void redirection (
    char * command,
    int new,
    int old )
```

Here is the caller graph for this function:

**4.114.1.18 rmdirs()**

```
void rmdirs (
    const char * path )
```

Here is the call graph for this function: Here is the caller graph for this function:

**4.114.1.19 score\_blank()**

```
int score_blank (
    char * id,
    char * filename )
```

Here is the call graph for this function: Here is the caller graph for this function:

**4.114.1.20 score\_program()**

```
double score_program (
    char * id,
    char * filename )
```

Here is the call graph for this function: Here is the caller graph for this function:

**4.114.1.21 score\_student()**

```
double score_student (
    int fd,
    char * id )
```

Here is the call graph for this function: Here is the caller graph for this function:

**4.114.1.22 score\_students()**

```
void score_students ( )
```

Here is the call graph for this function: Here is the caller graph for this function:

**4.114.1.23 set\_idTable()**

```
void set_idTable (
    char * stuDir )
```

Here is the call graph for this function: Here is the caller graph for this function:

**4.114.1.24 set\_scoreTable()**

```
void set_scoreTable (
    char * ansDir )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.114.1.25 sort\_idTable()

```
void sort_idTable (
    int size )
```

Here is the caller graph for this function:

#### 4.114.1.26 sort\_scoreTable()

```
void sort_scoreTable (
    int size )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.114.1.27 ssu\_score()

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

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.114.1.28 to\_lower\_case()

```
void to_lower_case (
    char * c )
```

Here is the caller graph for this function:

#### 4.114.1.29 write\_first\_row()

```
void write_first_row (
    int fd )
```

Here is the caller graph for this function:

#### 4.114.1.30 write\_scoreTable()

```
void write_scoreTable (
    char * filename )
```

Here is the caller graph for this function:

### 4.114.2 Variable Documentation

#### 4.114.2.1 ansDir

```
char ansDir[BUFLen]
```

#### 4.114.2.2 eOption

```
int eOption = false
```

```
-i
```

#### 4.114.2.3 errorDir

```
char errorDir[BUFLen]
```

#### 4.114.2.4 id\_table

```
char id_table
```

#### 4.114.2.5 iOption

```
int iOption = false
```

#### 4.114.2.6 mOption

```
int mOption = false
```

#### 4.114.2.7 printId

```
char printId[ARGNUM][FILELEN]
```



#### 4.114.2.8 score\_table

```
struct ssu_scoreTable score_table
```

#### 4.114.2.9 stuDir

```
char stuDir[BUFLLEN]
```

#### 4.114.2.10 threadFiles

```
char threadFiles[ARGNUM][FILELEN]
```

#### 4.114.2.11 tOption

```
int tOption = false
```

## 4.115 ssu\_score.h File Reference

This graph shows which files directly or indirectly include this file:

### Classes

- struct [ssu\\_scoreTable](#)

### Macros

- #define [ARGNUM](#) 5
- #define [BUFLLEN](#) 1024
- #define [CFILE](#) 4
- #define [ERROR](#) 0
- #define [false](#) 0
- #define [FILELEN](#) 64
- #define [OVER](#) 5
- #define [QNUM](#) 100
- #define [SNUM](#) 100
- #define [STDERR](#) 2
- #define [STDOUT](#) 1
- #define [TEXTFILE](#) 3
- #define [true](#) 1
- #define [WARNING](#) -0.1

## Functions

- double [check\\_error\\_warning](#) (char \*filename)
- int [check\\_option](#) (int argc, char \*argv[])
- int [compare\\_resultfile](#) (char \*file1, char \*file2)
- double [compile\\_program](#) (char \*id, char \*filename)
- void [do\\_iOption](#) (int argc, char \*argv[], int optind)
- int [execute\\_program](#) (char \*id, char \*filename)
- char \* [get\\_answer](#) (int fd, char \*result)
- int [get\\_create\\_type](#) ()
- int [get\\_file\\_type](#) (char \*filename)
- void [get\\_qname\\_number](#) (char \*qname, int \*num1, int \*num2)
- pid\_t [inBackground](#) (char \*name)
- int [is\\_exist](#) (char(\*src)[FILELEN], char \*target)
- int [is\\_thread](#) (char \*qname)
- void [make\\_scoreTable](#) (char \*ansDir)
- void [print\\_usage](#) ()
- void [read\\_scoreTable](#) (char \*path)
- void [redirection](#) (char \*command, int newfd, int oldfd)
- void [rmdirs](#) (const char \*path)
- int [score\\_blank](#) (char \*id, char \*filename)
- double [score\\_program](#) (char \*id, char \*filename)
- double [score\\_student](#) (int fd, char \*id)
- void [score\\_students](#) ()
- void [set\\_idTable](#) (char \*stuDir)
- void [set\\_scoreTable](#) (char \*ansDir)
- void [sort\\_idTable](#) (int size)
- void [sort\\_scoreTable](#) (int size)
- void [ssu\\_score](#) (int argc, char \*argv[])
- void [to\\_lower\\_case](#) (char \*c)
- void [write\\_first\\_row](#) (int fd)
- void [write\\_scoreTable](#) (char \*filename)

### 4.115.1 Macro Definition Documentation

#### 4.115.1.1 ARGNUM

```
#define ARGNUM 5
```

#### 4.115.1.2 BUFLen

```
#define BUFLen 1024
```

#### 4.115.1.3 CFILE

```
#define CFILE 4
```

#### 4.115.1.4 ERROR

```
#define ERROR 0
```

#### 4.115.1.5 false

```
#define false 0
```

#### 4.115.1.6 FILELEN

```
#define FILELEN 64
```

#### 4.115.1.7 OVER

```
#define OVER 5
```

#### 4.115.1.8 QNUM

```
#define QNUM 100
```

#### 4.115.1.9 SNUM

```
#define SNUM 100
```

#### 4.115.1.10 STDERR

```
#define STDERR 2
```

#### 4.115.1.11 STDOUT

```
#define STDOUT 1
```

#### 4.115.1.12 TEXTFILE

```
#define TEXTFILE 3
```

#### 4.115.1.13 true

```
#define true 1
```

#### 4.115.1.14 WARNING

```
#define WARNING -0.1
```

### 4.115.2 Function Documentation

#### 4.115.2.1 check\_error\_warning()

```
double check_error_warning (
    char * filename )
```

Here is the caller graph for this function:

#### 4.115.2.2 check\_option()

```
int check_option (
    int argc,
    char * argv[] )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.115.2.3 compare\_resultfile()

```
int compare_resultfile (
    char * file1,
    char * file2 )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.115.2.4 compile\_program()

```
double compile_program (
    char * id,
    char * filename )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.115.2.5 do\_iOption()

```
void do_iOption (
    int argc,
    char * argv[],
    int optind )
```

Here is the caller graph for this function:

#### 4.115.2.6 execute\_program()

```
int execute_program (
    char * id,
    char * filename )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.115.2.7 get\_answer()

```
char* get_answer (
    int fd,
    char * result )
```

Here is the caller graph for this function:

#### 4.115.2.8 get\_create\_type()

```
int get_create_type ( )
```

Here is the caller graph for this function:

#### 4.115.2.9 get\_file\_type()

```
int get_file_type (
    char * filename )
```

Here is the caller graph for this function:

#### 4.115.2.10 get\_qname\_number()

```
void get_qname_number (
    char * qname,
    int * num1,
    int * num2 )
```

Here is the caller graph for this function:

#### 4.115.2.11 inBackground()

```
pid_t inBackground (
    char * name )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.115.2.12 is\_exist()

```
int is_exist (
    char(*) src[FILELEN],
    char * target )
```

#### 4.115.2.13 is\_thread()

```
int is_thread (
    char * qname )
```

Here is the caller graph for this function:

#### 4.115.2.14 make\_scoreTable()

```
void make_scoreTable (
    char * ansDir )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.115.2.15 print\_usage()

```
void print_usage ( )
```

Here is the caller graph for this function:

#### 4.115.2.16 read\_scoreTable()

```
void read_scoreTable (
    char * path )
```

Here is the caller graph for this function:

#### 4.115.2.17 redirection()

```
void redirection (
    char * command,
    int newfd,
    int oldfd )
```

Here is the caller graph for this function:

#### 4.115.2.18 rmdirs()

```
void rmdirs (
    const char * path )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.115.2.19 score\_blank()

```
int score_blank (
    char * id,
    char * filename )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.115.2.20 score\_program()

```
double score_program (
    char * id,
    char * filename )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.115.2.21 score\_student()

```
double score_student (
    int fd,
    char * id )
```

Here is the call graph for this function: Here is the caller graph for this function:

#### 4.115.2.22 score\_students()

```
void score_students ( )
```

Here is the call graph for this function: Here is the caller graph for this function:

**4.115.2.23 set\_idTable()**

```
void set_idTable (
    char * stuDir )
```

Here is the call graph for this function: Here is the caller graph for this function:

**4.115.2.24 set\_scoreTable()**

```
void set_scoreTable (
    char * ansDir )
```

Here is the call graph for this function: Here is the caller graph for this function:

**4.115.2.25 sort\_idTable()**

```
void sort_idTable (
    int size )
```

Here is the caller graph for this function:

**4.115.2.26 sort\_scoreTable()**

```
void sort_scoreTable (
    int size )
```

Here is the call graph for this function: Here is the caller graph for this function:

**4.115.2.27 ssu\_score()**

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

Here is the call graph for this function: Here is the caller graph for this function:

**4.115.2.28 to\_lower\_case()**

```
void to_lower_case (
    char * c )
```

Here is the caller graph for this function:

**4.115.2.29 write\_first\_row()**

```
void write_first_row (
    int fd )
```

Here is the caller graph for this function:

**4.115.2.30 write\_scoreTable()**

```
void write_scoreTable (
    char * filename )
```

Here is the caller graph for this function: