

Minimal Disks Simulation

Generated by Doxygen 1.13.2

1 Class Index	1
1.1 Class List	1
2 File Index	3
2.1 File List	3
3 Class Documentation	5
3.1 Disk Class Reference	5
3.1.1 Detailed Description	5
3.1.2 Constructor & Destructor Documentation	5
3.1.2.1 Disk()	5
3.1.3 Member Function Documentation	6
3.1.3.1 distance()	6
3.1.3.2 move()	6
3.1.4 Member Data Documentation	6
3.1.4.1 radius	6
3.1.4.2 x	6
3.1.4.3 y	6
3.2 System Class Reference	6
3.2.1 Detailed Description	7
3.2.2 Constructor & Destructor Documentation	7
3.2.2.1 System()	7
3.2.3 Member Function Documentation	7
3.2.3.1 enforceBoundaries()	7
3.2.3.2 overlap()	7
3.2.3.3 save()	7
3.2.3.4 step()	7
3.2.3.5 uniform()	8
3.2.4 Member Data Documentation	8
3.2.4.1 boxSize	8
3.2.4.2 disks	8
3.2.4.3 displacement	8
3.2.4.4 dist	8
3.2.4.5 gen	8
4 File Documentation	9
4.1 disk.cpp File Reference	9
4.2 disk.h File Reference	10
4.3 disk.h	10
4.4 main.cpp File Reference	10
4.4.1 Function Documentation	11
4.4.1.1 main()	11
4.5 system.cpp File Reference	11

4.6 system.h File Reference	12
4.7 system.h	12
Index	15

Chapter 1

Class Index

1.1 Class List

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

Disk	A simple class representing hard disks	5
System	A simple class containing the system's parameters	6

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

disk.cpp	9
disk.h	10
main.cpp	10
system.cpp	11
system.h	12

Chapter 3

Class Documentation

3.1 Disk Class Reference

A simple class representing hard disks.

```
#include <disk.h>
```

Public Member Functions

- [Disk](#) (double [x](#), double [y](#), double [r](#))
- void [move](#) (double [dx](#), double [dy](#))
- double [distance](#) ([Disk](#) &[d](#))

Public Attributes

- double [x](#)
- double [y](#)
- double [radius](#)

3.1.1 Detailed Description

A simple class representing hard disks.

3.1.2 Constructor & Destructor Documentation

3.1.2.1 Disk()

```
Disk::Disk (  
    double x,  
    double y,  
    double r)
```

3.1.3 Member Function Documentation

3.1.3.1 distance()

```
double Disk::distance (  
    Disk & d)
```

3.1.3.2 move()

```
void Disk::move (  
    double dx,  
    double dy)
```

3.1.4 Member Data Documentation

3.1.4.1 radius

```
double Disk::radius
```

3.1.4.2 x

```
double Disk::x
```

3.1.4.3 y

```
double Disk::y
```

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

- [disk.h](#)
- [disk.cpp](#)

3.2 System Class Reference

A simple class containing the system's parameters.

```
#include <system.h>
```

Public Member Functions

- [System](#) (int N, double [displacement](#), double radius, double box_size, int seed)
- void [step](#) ()
- void [save](#) (const std::string &filename)
- bool [overlap](#) (int i)
- void [enforceBoundaries](#) ([Disk](#) &disk)
- double [uniform](#) (double min, double max)

Public Attributes

- `std::vector< Disk > disks`
- `double boxSize`
- `double displacement`
- `std::mt19937 gen`
- `std::uniform_real_distribution< double > dist`

3.2.1 Detailed Description

A simple class containing the system's parameters.

3.2.2 Constructor & Destructor Documentation

3.2.2.1 System()

```
System::System (  
    int N,  
    double displacement,  
    double radius,  
    double box_size,  
    int seed)
```

3.2.3 Member Function Documentation

3.2.3.1 enforceBoundaries()

```
void System::enforceBoundaries (  
    Disk & disk)
```

3.2.3.2 overlap()

```
bool System::overlap (  
    int i)
```

3.2.3.3 save()

```
void System::save (  
    const std::string & filename)
```

3.2.3.4 step()

```
void System::step ()
```

3.2.3.5 uniform()

```
double System::uniform (
    double min,
    double max)
```

3.2.4 Member Data Documentation

3.2.4.1 boxSize

```
double System::boxSize
```

3.2.4.2 disks

```
std::vector<Disk> System::disks
```

3.2.4.3 displacement

```
double System::displacement
```

3.2.4.4 dist

```
std::uniform_real_distribution<double> System::dist
```

3.2.4.5 gen

```
std::mt19937 System::gen
```

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

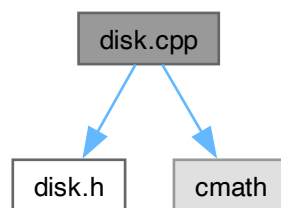
- [system.h](#)
- [system.cpp](#)

Chapter 4

File Documentation

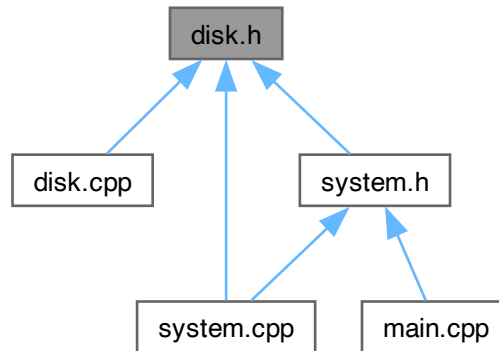
4.1 disk.cpp File Reference

```
#include "disk.h"  
#include <cmath>  
Include dependency graph for disk.cpp:
```



4.2 disk.h File Reference

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



Classes

- class [Disk](#)
A simple class representing hard disks.

4.3 disk.h

[Go to the documentation of this file.](#)

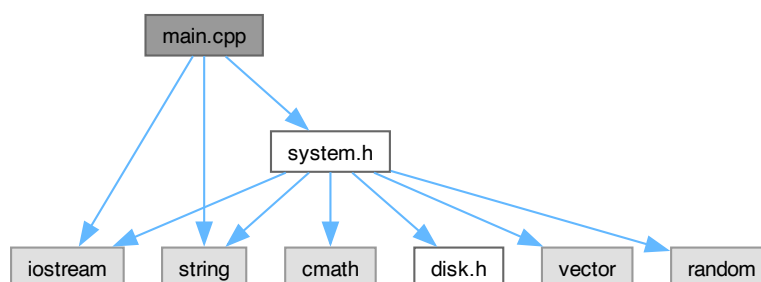
```
00001
00002 #ifndef DISK_H
00003 #define DISK_H
00004
00009 class Disk {
00010 public:
00011     double x, y, radius;
00012     Disk(double x, double y, double r) ;
00013
00014     void move(double dx, double dy) ;
00015     double distance(Disk& d) ;
00016 };
00017
00018 #endif // !DISK_H
```

4.4 main.cpp File Reference

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

```
#include "system.h"
```

Include dependency graph for main.cpp:



Functions

- `int main()`

Entry point of the program.

4.4.1 Function Documentation

4.4.1.1 main()

```
int main ()
```

Entry point of the program.

This function initializes the program and manages execution.

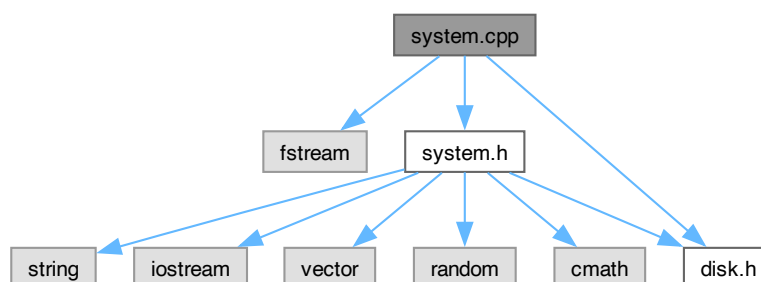
4.5 system.cpp File Reference

```
#include <fstream>
```

```
#include "system.h"
```

```
#include "disk.h"
```

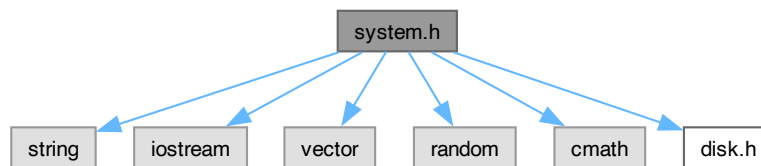
Include dependency graph for system.cpp:



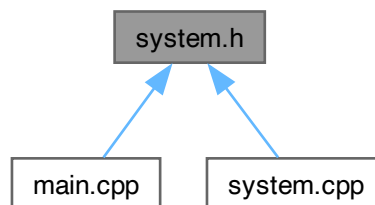
4.6 system.h File Reference

```
#include <string>
#include <iostream>
#include <vector>
#include <random>
#include <cmath>
#include "disk.h"
```

Include dependency graph for system.h:



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



Classes

- class [System](#)

A simple class containing the system's parameters.

4.7 system.h

[Go to the documentation of this file.](#)

```
00001 #ifndef SYSTEM_H
00002 #define SYSTEM_H
00003
00004 #include <string>
00005 #include <iostream>
00006 #include <vector>
00007 #include <random>
```



```
00008 #include <cmath>
00009 #include "disk.h"
00010
00015 class System {
00016 public:
00017     std::vector<Disk> disks;
00018     double boxSize;
00019     double displacement;
00020     std::mt19937 gen;
00021     std::uniform_real_distribution<double> dist;
00022
00023     System(int N, double displacement, double radius, double box_size, int seed);
00024
00025     void step() ;
00026
00027     void save(const std::string &filename);
00028
00029     bool overlap(int i);
00030
00031     void enforceBoundaries(Disk & disk) ;
00032     double uniform(double min, double max);
00033 };
00034 #endif // !SYSTEM_H
```


Index

- boxSize
 - System, [8](#)
- Disk, [5](#)
 - Disk, [5](#)
 - distance, [6](#)
 - move, [6](#)
 - radius, [6](#)
 - x, [6](#)
 - y, [6](#)
- disk.cpp, [9](#)
- disk.h, [10](#)
- disks
 - System, [8](#)
- displacement
 - System, [8](#)
- dist
 - System, [8](#)
- distance
 - Disk, [6](#)
- enforceBoundaries
 - System, [7](#)
- gen
 - System, [8](#)
- main
 - main.cpp, [11](#)
- main.cpp, [10](#)
 - main, [11](#)
- move
 - Disk, [6](#)
- overlap
 - System, [7](#)
- radius
 - Disk, [6](#)
- save
 - System, [7](#)
- step
 - System, [7](#)
- System, [6](#)
 - boxSize, [8](#)
 - disks, [8](#)
 - displacement, [8](#)
 - dist, [8](#)
 - enforceBoundaries, [7](#)
 - gen, [8](#)
 - overlap, [7](#)
 - save, [7](#)
 - step, [7](#)
 - System, [7](#)
 - uniform, [7](#)
- system.cpp, [11](#)
- system.h, [12](#)
- uniform
 - System, [7](#)
- x
 - Disk, [6](#)
- y
 - Disk, [6](#)