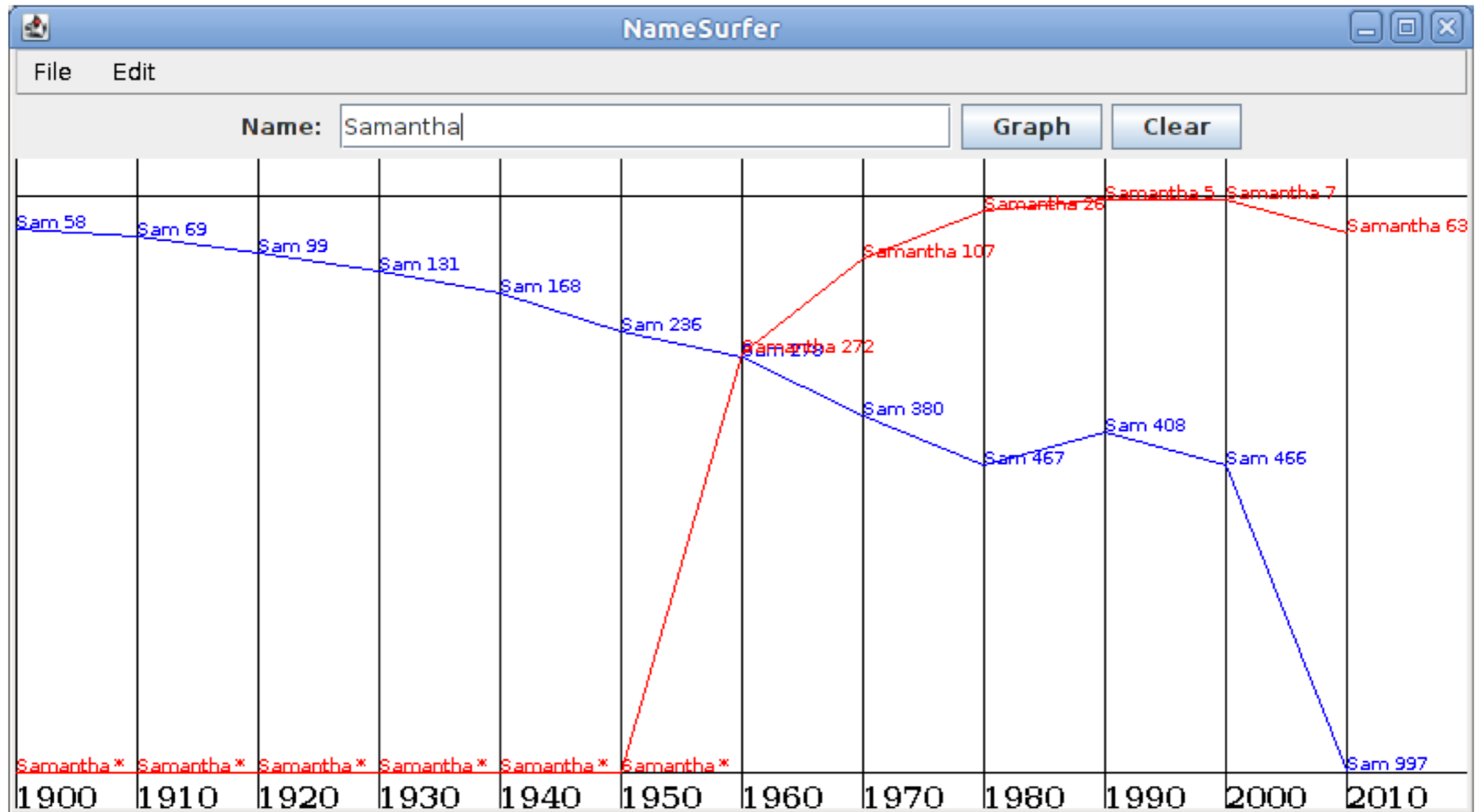


YEAH session #6

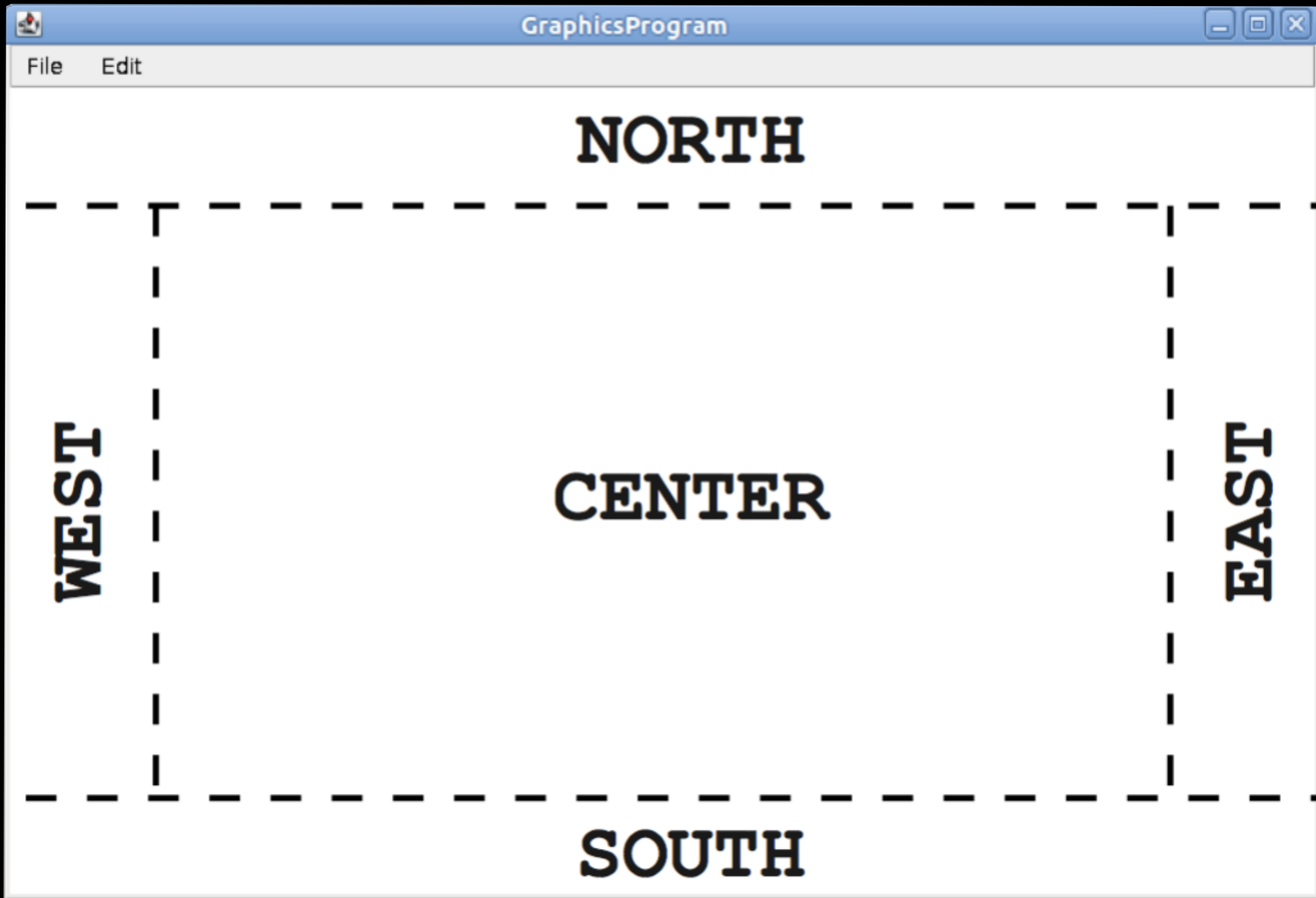


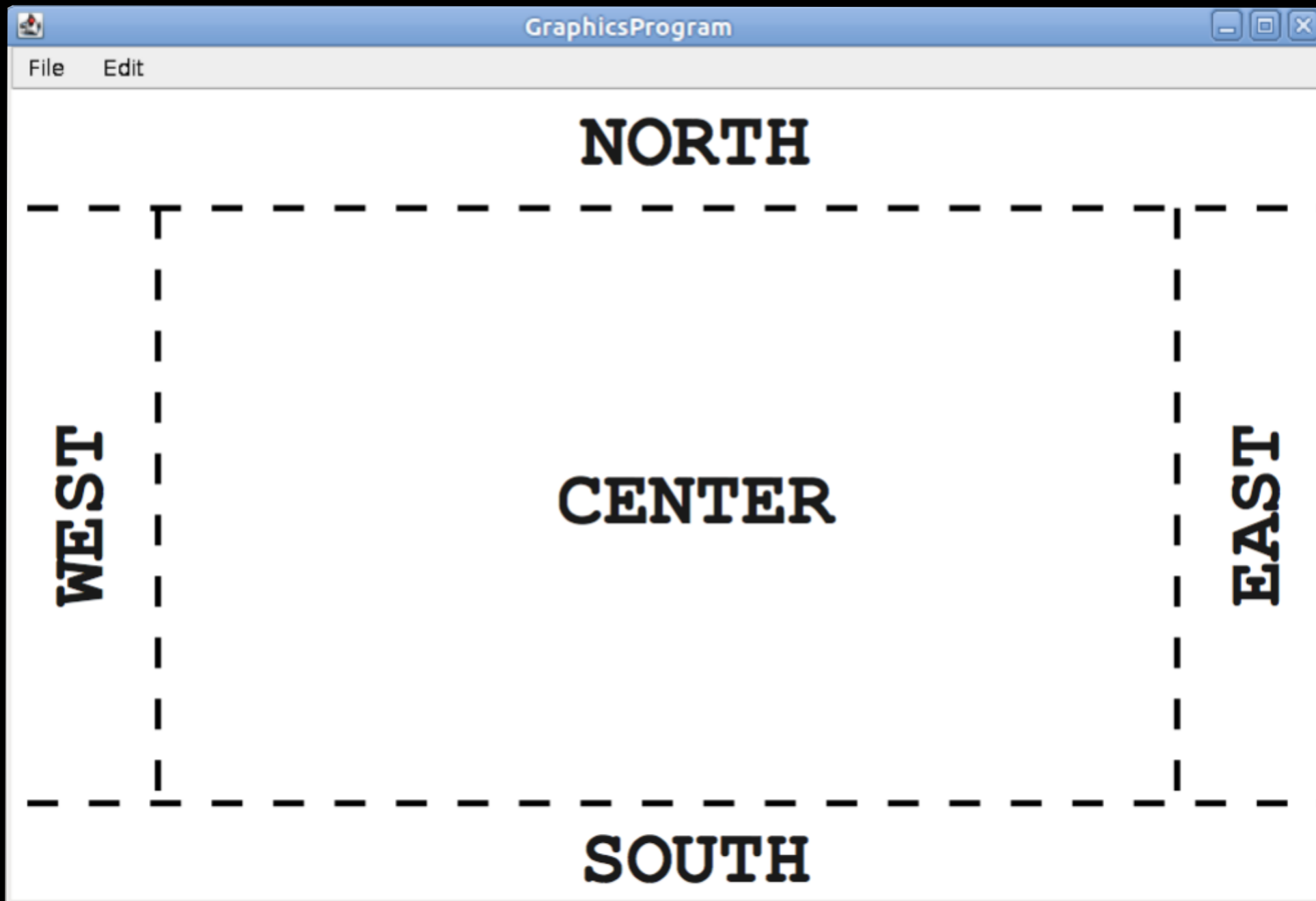
6 March 2014, 5:30p-6:30p
Miles Seiver

Review session schedule

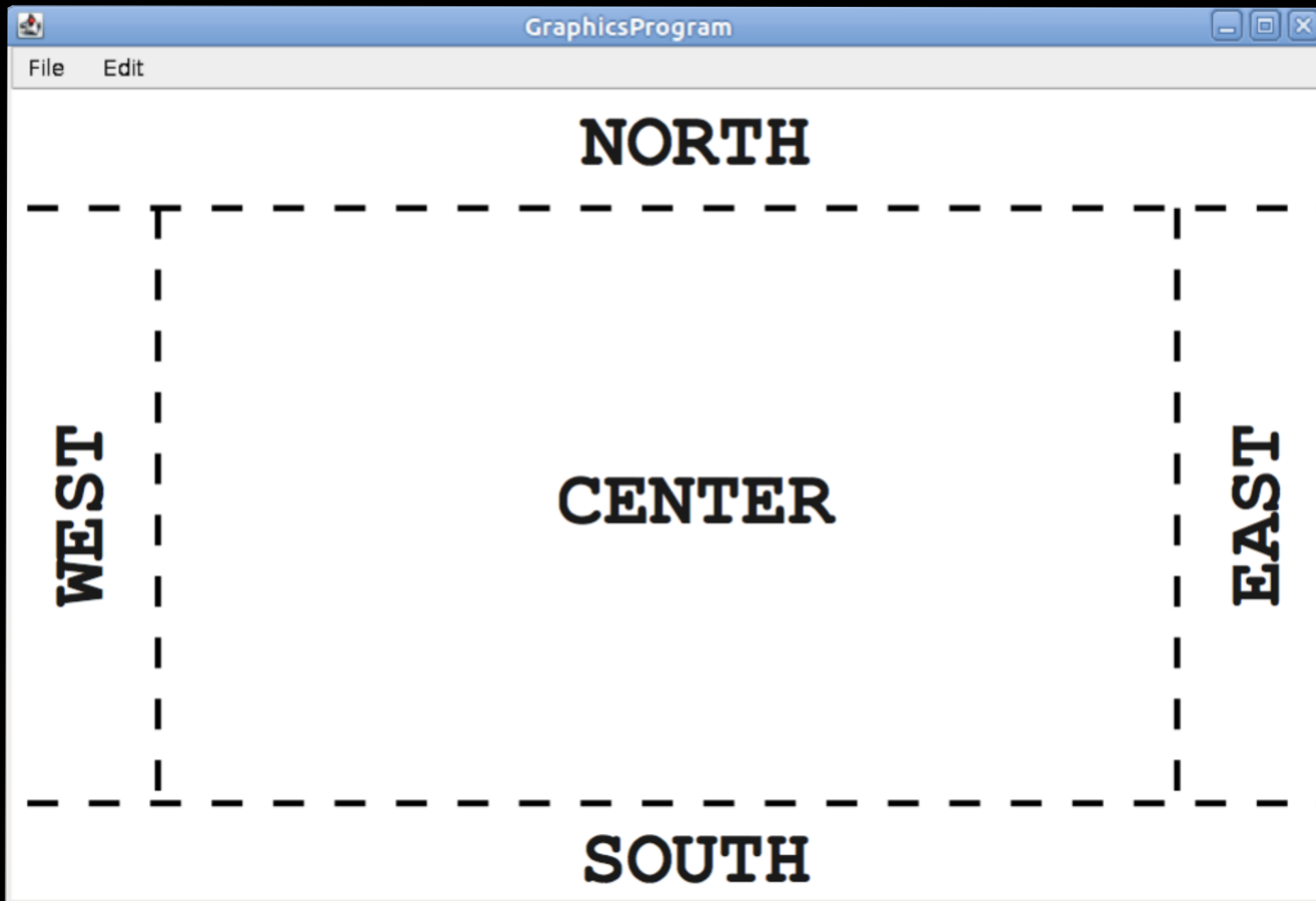
Topic	Date	Time	Location
assignment 6	today!	now!	here!
assignment 7	Sun 16 Mar	7p - 8p	Hewlett 200

Interactors

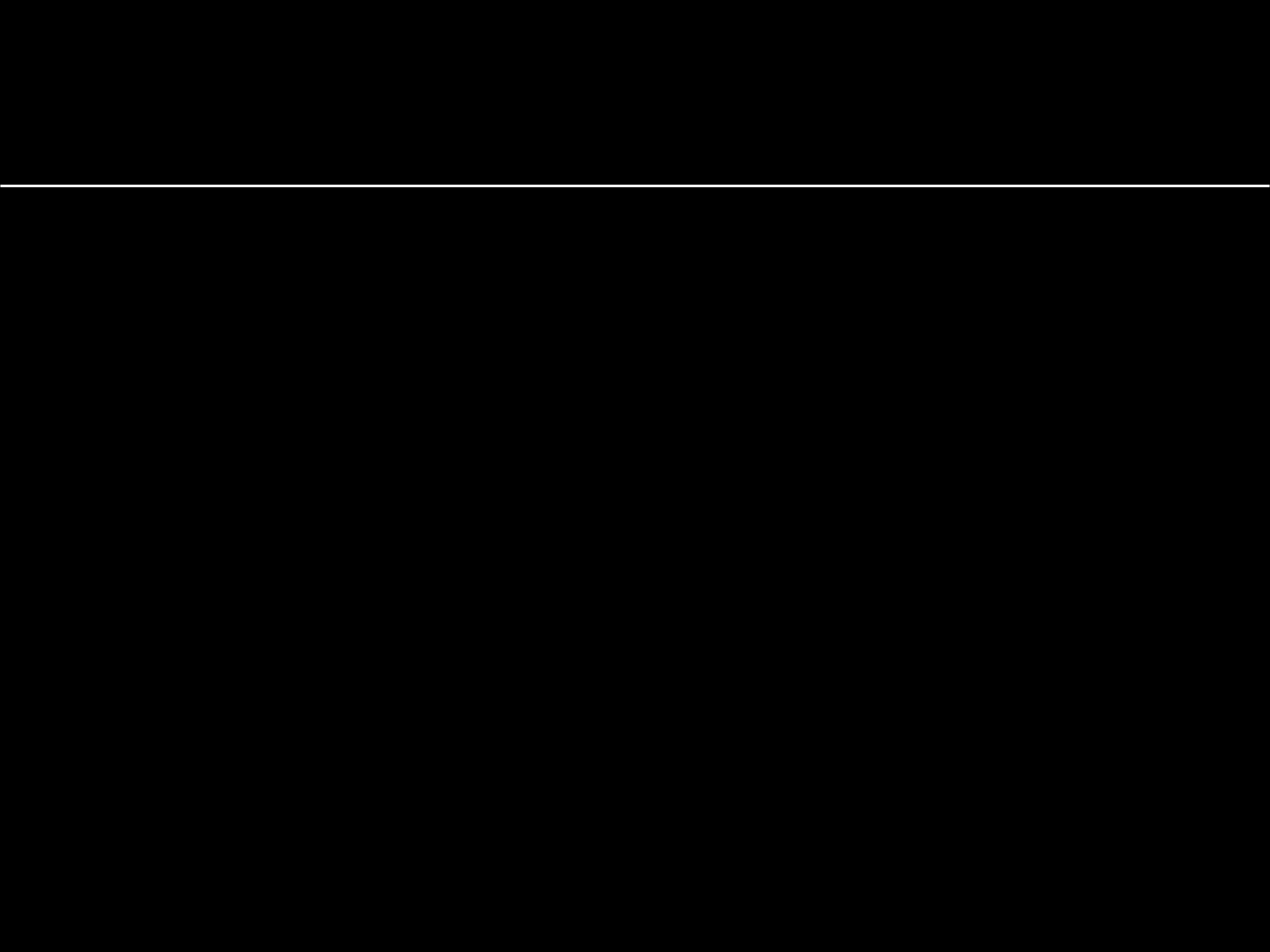




```
JSlider slide = new JSlider(min, max, initial);  
JButton button = new JButton("[button text]");  
JTextField field = new JTextField("[field text]");
```



```
JSlider slide = new JSlider(min, max, initial);  
JButton button = new JButton("[button text]");  
JTextField field = new JTextField("[field text]");  
add(interactor, location);
```



```
addActionListeners();
```

```
addActionListeners();
```

```
public void actionPerformed(ActionEvent e) {
```

```
}
```

```
addActionListeners();
```

```
public void actionPerformed(ActionEvent e) {  
    if (e.getSource() == someInteractorIvar) {  
        . . .  
    }  
  
}
```

```
addActionListeners();
```

```
public void actionPerformed(ActionEvent e) {  
    if (e.getSource() == someInteractorIvar) {  
        . . .  
    }  
  
}
```

requires instance variables

```
addActionListeners();
```

```
public void actionPerformed(ActionEvent e) {  
    if (e.getSource() == someInteractorIvar) {  
        . . .  
    }  
  
    if (e.getActionCommand().equals("[text in button]")) {  
        . . .  
    }  
  
}
```

```
addActionListeners();
```

```
public void actionPerformed(ActionEvent e) {
```

```
    if (e.getSource() == someInteractorIvar) {
```

```
        . . .
```

requires instance variables

```
    }
```

```
    if (e.getActionCommand().equals("[text in button]")) {
```

```
        . . .
```

```
    }
```

only works for buttons unless you use

```
someInteractor.setActionCommand("[action command]");
```

```
}
```

the assignment

NameSurfer

due Wed, 12 Mar @ 3:15pm



demo

New Java tools you will use!

New Java tools you will use!

HashMap

New Java tools you will use!

HashMap

Array

New Java tools you will use!

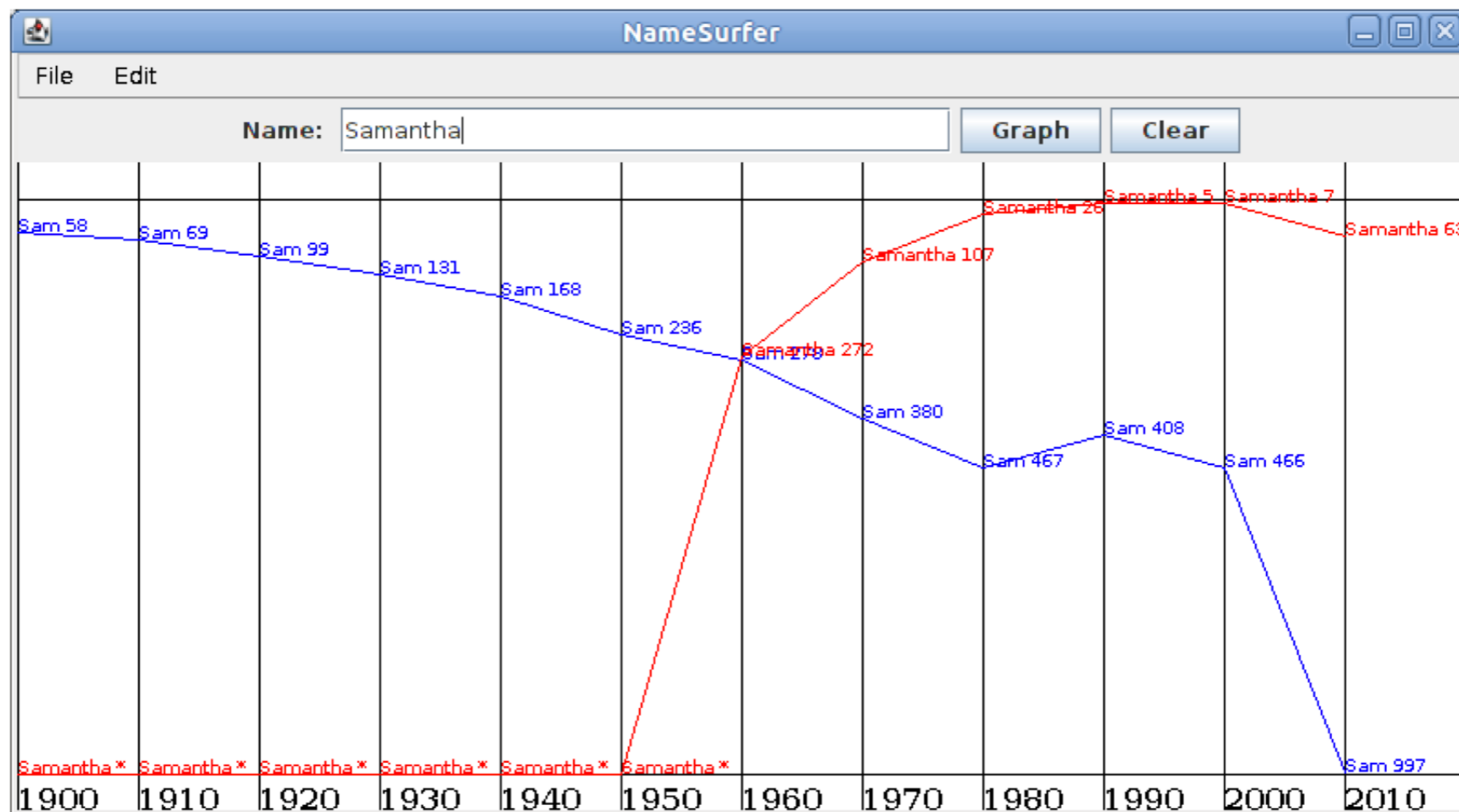
HashMap

Array

Interactors

NamesData.txt

```
...  
Sam 58 69 99 131 168 236 278 380 467 408 466 997  
Samantha 0 0 0 0 0 0 272 107 26 5 7 63  
Samara 0 0 0 0 0 0 0 0 0 0 886 0  
Samir 0 0 0 0 0 0 0 920 0 798 0  
Sammie 537 545 351 325 333 396 565 772 930 0 0 0  
Sammy 0 887 544 299 202 262 321 395 575 639 755 0  
Samson 0 0 0 0 0 0 0 0 0 0 915 0  
Samuel 31 41 46 60 61 71 83 61 52 35 28 32  
Sandi 0 0 0 0 704 864 621 695 0 0 0 0  
Sandra 0 942 606 50 6 12 11 39 94 168 257 962  
...
```



NameSurferDataBase

Loads and manages
NameSurferEntryS

NameSurferDataBase

Loads and manages
NameSurferEntryS

NameSurferEntry

NameSurferEntry

NameSurferEntry

NamesData.txt

```
...  
Sam 58 69 99 131 168 236 278 380 467 408 466  
Samantha 0 0 0 0 0 0 272 107 26 5 7  
Samara 0 0 0 0 0 0 0 0 0 0 886  
Samir 0 0 0 0 0 0 0 0 920 0 798  
Sammie 537 545 351 325 333 396 565 772 930 0 0  
Sammy 0 887 544 299 202 262 321 395 575 639 755  
Samson 0 0 0 0 0 0 0 0 0 0 915  
Samuel 31 41 46 60 61 71 83 61 52 35 28  
Sandi 0 0 0 0 704 864 621 695 0 0 0  
Sandra 0 942 606 50 6 12 11 39 94 168 257  
...
```

NameSurferDataBase

Loads and manages
NameSurferEntryS

NameSurferEntry

NameSurferEntry

NameSurferEntry

NamesData.txt

```
...  
Sam 58 69 99 131 168 236 278 380 467 408 466  
Samantha 0 0 0 0 0 0 272 107 26 5 7  
Samara 0 0 0 0 0 0 0 0 0 0 886  
Samir 0 0 0 0 0 0 0 0 920 0 798  
Sammie 537 545 351 325 333 396 565 772 930 0 0  
Sammy 0 887 544 299 202 262 321 395 575 639 755  
Samson 0 0 0 0 0 0 0 0 0 0 915  
Samuel 31 41 46 60 61 71 83 61 52 35 28  
Sandi 0 0 0 0 704 864 621 695 0 0 0  
Sandra 0 942 606 50 6 12 11 39 94 168 257  
...
```

NameSurferDataBase

Loads and manages
NameSurferEntryS

NameSurferEntry

NameSurferEntry

NameSurferEntry

NamesData.txt

```
...  
Sam 58 69 99 131 168 236 278 380 467 408 466  
Samantha 0 0 0 0 0 0 272 107 26 5 7  
Samara 0 0 0 0 0 0 0 0 0 0 886  
Samir 0 0 0 0 0 0 0 0 920 0 798  
Sammie 537 545 351 325 333 396 565 772 930 0 0  
Sammy 0 887 544 299 202 262 321 395 575 639 755  
Samson 0 0 0 0 0 0 0 0 0 0 915  
Samuel 31 41 46 60 61 71 83 61 52 35 28  
Sandi 0 0 0 0 704 864 621 695 0 0 0  
Sandra 0 942 606 50 6 12 11 39 94 168 257  
...
```

NameSurferDataBase

Loads and manages
NameSurferEntryS

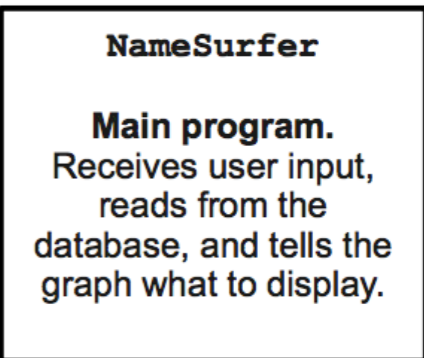
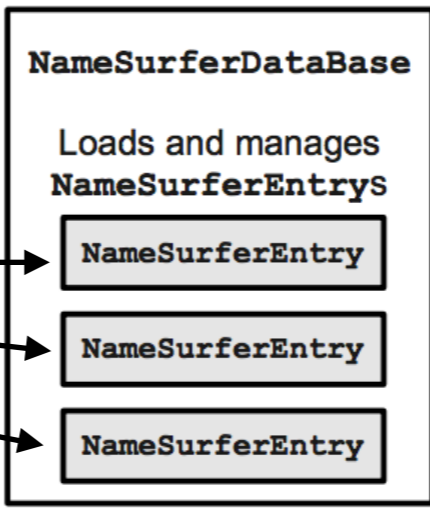
NameSurferEntry

NameSurferEntry

NameSurferEntry

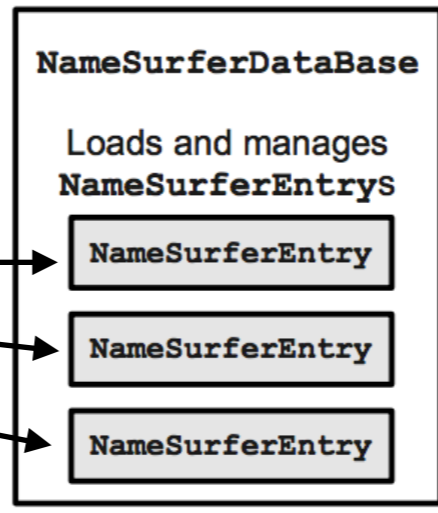
NamesData.txt

```
...  
Sam 58 69 99 131 168 236 278 380 467 408 466  
Samantha 0 0 0 0 0 0 272 107 26 5 7  
Samara 0 0 0 0 0 0 0 0 0 0 886  
Samir 0 0 0 0 0 0 0 0 920 0 798  
Sammie 537 545 351 325 333 396 565 772 930 0 0  
Sammy 0 887 544 299 202 262 321 395 575 639 755  
Samson 0 0 0 0 0 0 0 0 0 0 915  
Samuel 31 41 46 60 61 71 83 61 52 35 28  
Sandi 0 0 0 0 704 864 621 695 0 0 0  
Sandra 0 942 606 50 6 12 11 39 94 168 257  
...
```

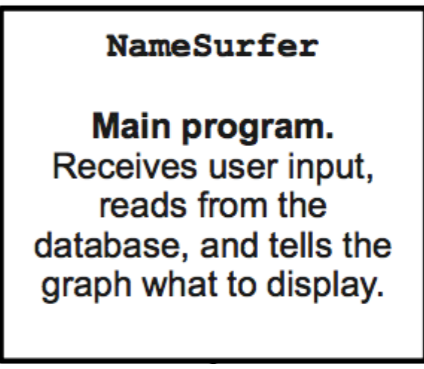


NamesData.txt

```
...  
Sam 58 69 99 131 168 236 278 380 467 408 466  
Samantha 0 0 0 0 0 0 272 107 26 5 7  
Samara 0 0 0 0 0 0 0 0 0 0 886  
Samir 0 0 0 0 0 0 0 0 920 0 798  
Sammie 537 545 351 325 333 396 565 772 930 0 0  
Sammy 0 887 544 299 202 262 321 395 575 639 755  
Samson 0 0 0 0 0 0 0 0 0 0 915  
Samuel 31 41 46 60 61 71 83 61 52 35 28  
Sandi 0 0 0 0 704 864 621 695 0 0 0  
Sandra 0 942 606 50 6 12 11 39 94 168 257  
...
```

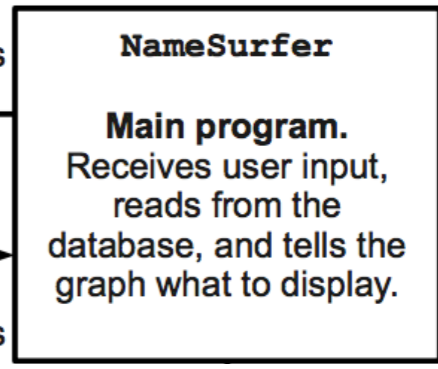
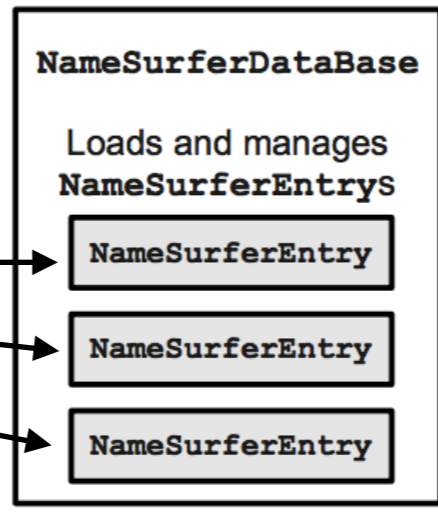


Asks for
NameSurferEntrys



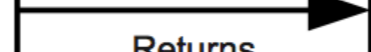
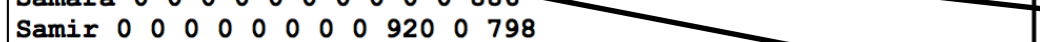
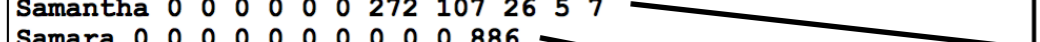
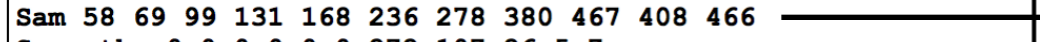
NamesData.txt

```
...  
Sam 58 69 99 131 168 236 278 380 467 408 466  
Samantha 0 0 0 0 0 0 272 107 26 5 7  
Samara 0 0 0 0 0 0 0 0 0 0 886  
Samir 0 0 0 0 0 0 0 0 920 0 798  
Sammie 537 545 351 325 333 396 565 772 930 0 0  
Sammy 0 887 544 299 202 262 321 395 575 639 755  
Samson 0 0 0 0 0 0 0 0 0 0 915  
Samuel 31 41 46 60 61 71 83 61 52 35 28  
Sandi 0 0 0 0 704 864 621 695 0 0 0  
Sandra 0 942 606 50 6 12 11 39 94 168 257  
...
```



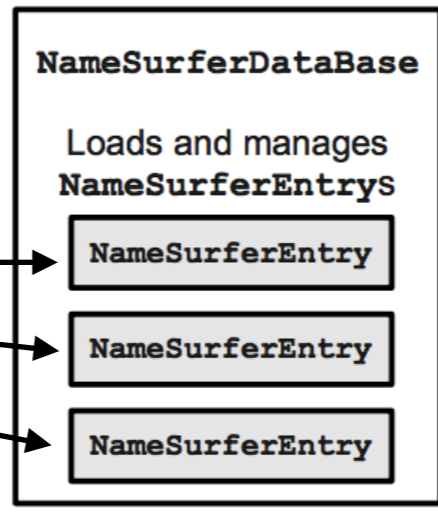
Asks for
NameSurferEntrys

Returns
NameSurferEntrys

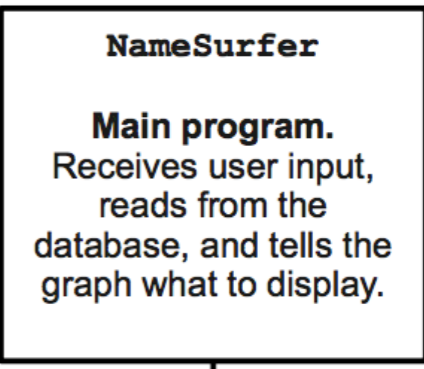


NamesData.txt

```
...  
Sam 58 69 99 131 168 236 278 380 467 408 466  
Samantha 0 0 0 0 0 0 272 107 26 5 7  
Samara 0 0 0 0 0 0 0 0 0 0 886  
Samir 0 0 0 0 0 0 0 0 920 0 798  
Sammie 537 545 351 325 333 396 565 772 930 0 0  
Sammy 0 887 544 299 202 262 321 395 575 639 755  
Samson 0 0 0 0 0 0 0 0 0 0 915  
Samuel 31 41 46 60 61 71 83 61 52 35 28  
Sandi 0 0 0 0 704 864 621 695 0 0 0  
Sandra 0 942 606 50 6 12 11 39 94 168 257  
...
```

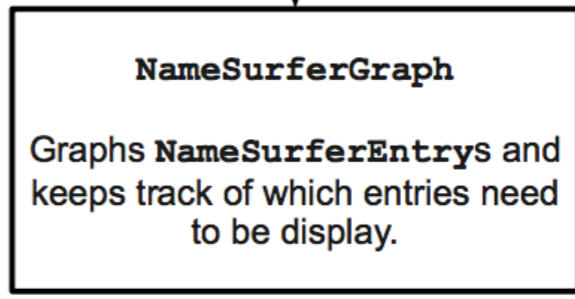


Asks for NameSurferEntry



Returns NameSurferEntry

Hands over NameSurferEntry that need to be displayed.

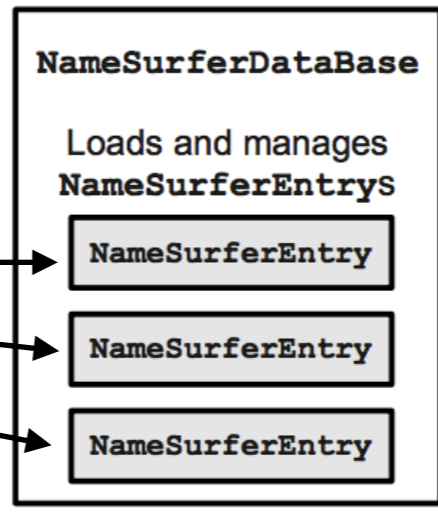


NamesData.txt

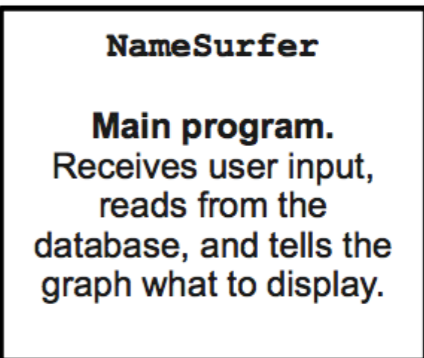
```

...
Sam 58 69 99 131 168 236 278 380 467 408 466
Samantha 0 0 0 0 0 0 272 107 26 5 7
Samara 0 0 0 0 0 0 0 0 0 0 886
Samir 0 0 0 0 0 0 0 0 920 0 798
Sammie 537 545 351 325 333 396 565 772 930 0 0
Sammy 0 887 544 299 202 262 321 395 575 639 755
Samson 0 0 0 0 0 0 0 0 0 0 915
Samuel 31 41 46 60 61 71 83 61 52 35 28
Sandi 0 0 0 0 704 864 621 695 0 0 0
Sandra 0 942 606 50 6 12 11 39 94 168 257
...

```

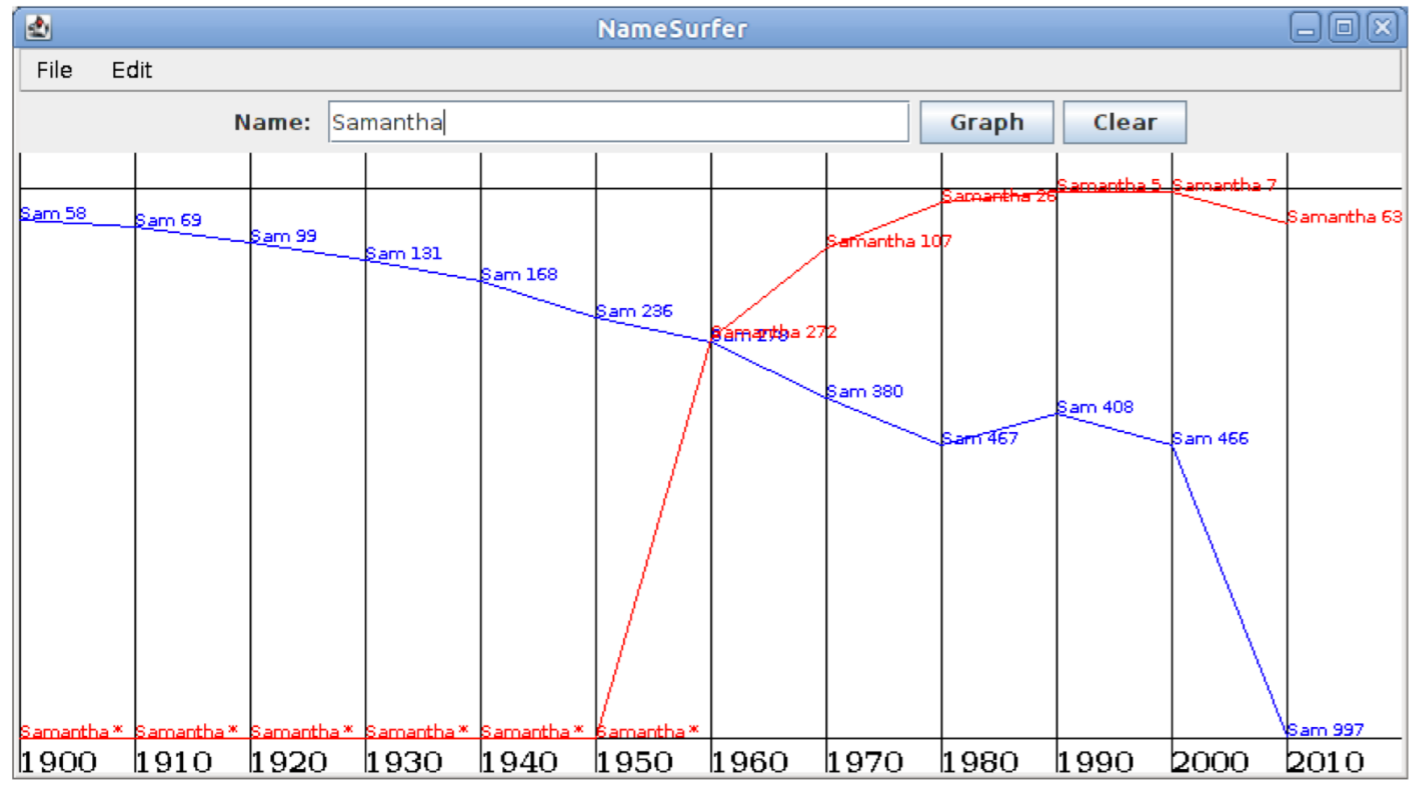
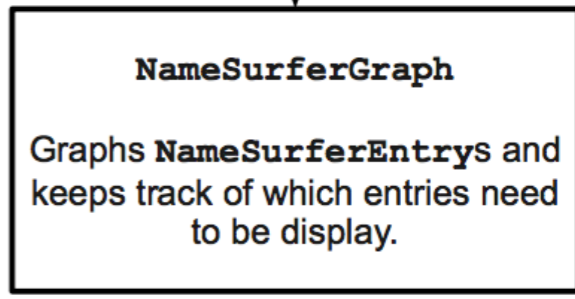


Asks for NameSurferEntry

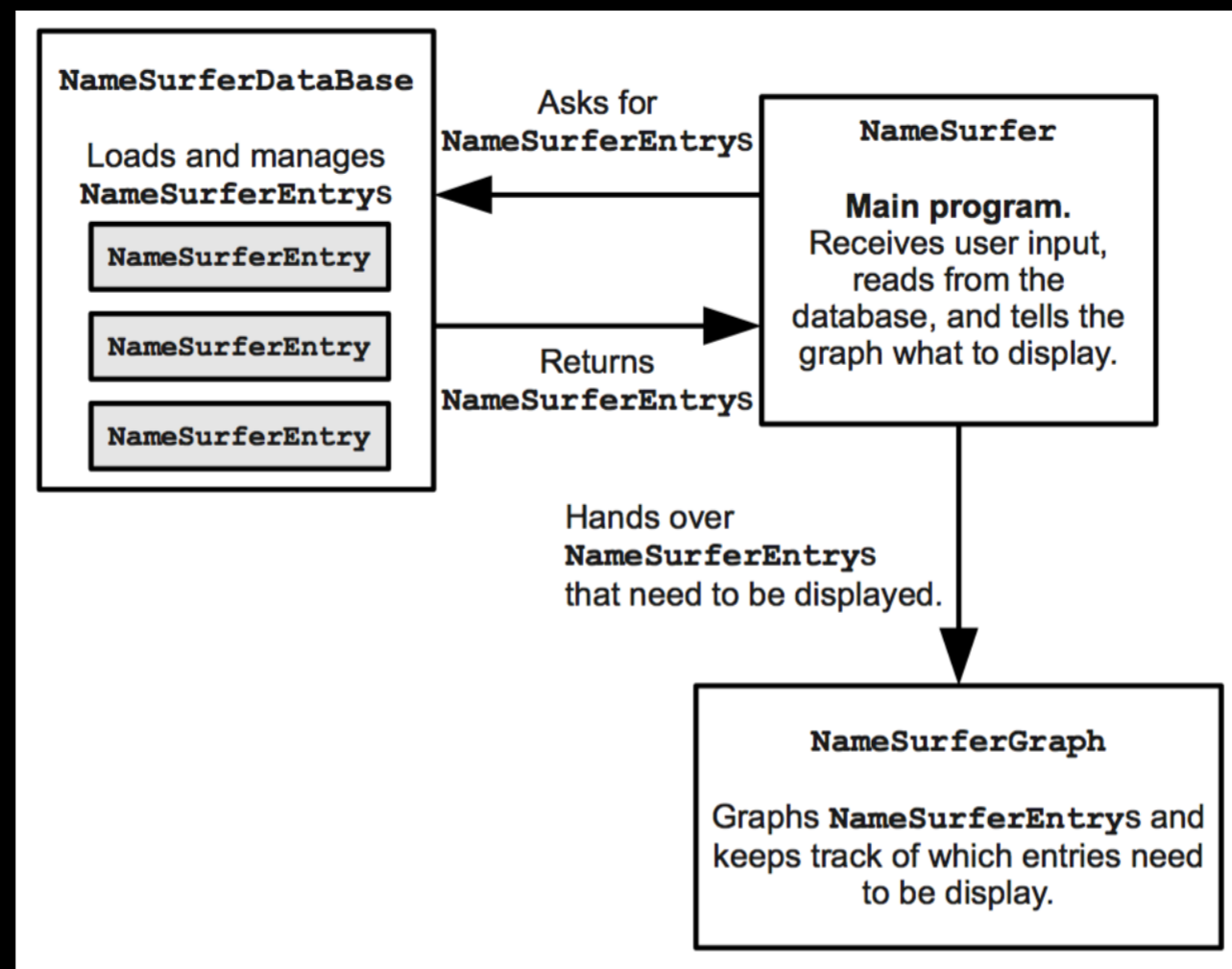


Returns NameSurferEntry

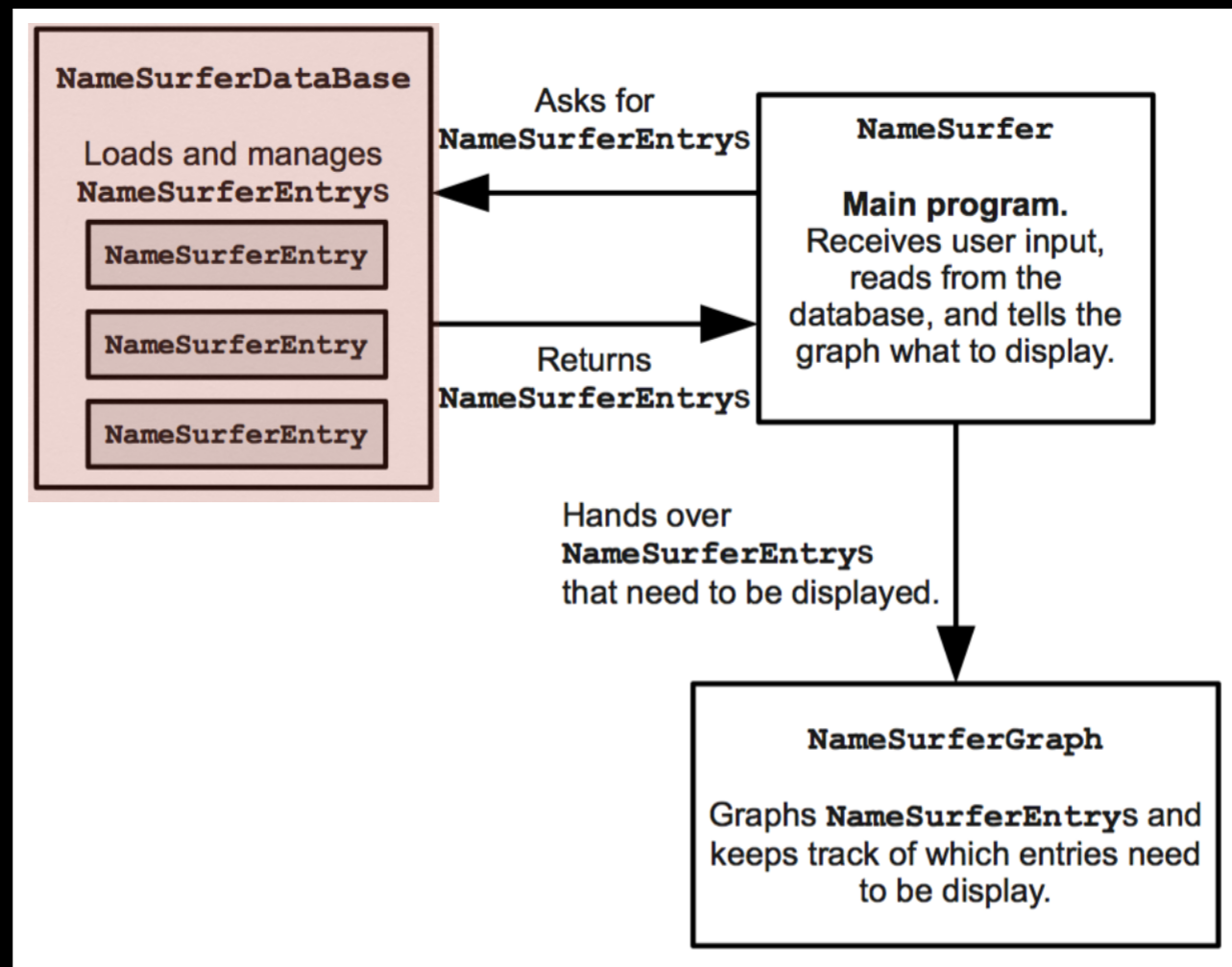
Hands over NameSurferEntry that need to be displayed.



NameSurferDatabase



NameSurferDatabase



```
public class NameSurferDataBase implements NameSurferConstants {

    /* Constructor: NameSurferDataBase(filename) */
    /**
     * Creates a new NameSurferDataBase and initializes it using the
     * data in the specified file. The constructor throws an error
     * exception if the requested file does not exist or if an error
     * occurs as the file is being read.
     */
    public NameSurferDataBase(String filename) {
        // You fill this in //
    }

    /* Method: findEntry(name) */
    /**
     * Returns the NameSurferEntry associated with this name, if one
     * exists. If the name does not appear in the database, this
     * method returns null.
     */
    public NameSurferEntry findEntry(String name) {
        // You need to turn this stub into a real implementation //
        return null;
    }
}
```

```
public class NameSurferDataBase implements NameSurferConstants {
```

```
/* Constructor: NameSurferDataBase(filename) */
```

```
/**
```

```
* Creates a new NameSurferDataBase and initializes it using the  
* data in the specified file. The constructor throws an error  
* exception if the requested file does not exist or if an error  
* occurs as the file is being read.
```

```
*/
```

```
public NameSurferDataBase(String filename) {  
    // You fill this in //  
}
```

```
/* Method: findEntry(name) */
```

```
/**
```

```
* Returns the NameSurferEntry associated with this name, if one  
* exists. If the name does not appear in the database, this  
* method returns null.
```

```
*/
```

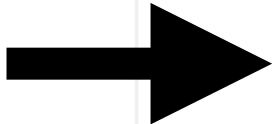
```
public NameSurferEntry findEntry(String name) {  
    // You need to turn this stub into a real implementation //  
    return null;  
}
```

```
}
```

```
public class NameSurferDataBase implements NameSurferConstants {

    /* Constructor: NameSurferDataBase(filename) */
    /**
     * Creates a new NameSurferDataBase and initializes it using the
     * data in the specified file. The constructor throws an error
     * exception if the requested file does not exist or if an error
     * occurs as the file is being read.
     */
    public NameSurferDataBase(String filename) {
        // You fill this in //
    }

    /* Method: findEntry(name) */
    /**
     * Returns the NameSurferEntry associated with this name, if one
     * exists. If the name does not appear in the database, this
     * method returns null.
     */
    public NameSurferEntry findEntry(String name) {
        // You need to turn this stub into a real implementation //
        return null;
    }
}
```



```
public class NameSurferDataBase implements NameSurferConstants {
```

```
/* Constructor: NameSurferDataBase(filename) */
```

```
/**
```

```
* Creates a new NameSurferDataBase and initializes it using the  
* data in the specified file. The constructor throws an error  
* exception if the requested file does not exist or if an error  
* occurs as the file is being read.
```

```
*/
```

```
public NameSurferDataBase(String filename) {  
    // You fill this in //  
}
```

name is case-insensitive


```
/* Method: findEntry(name) */
```

```
/**
```

```
* Returns the NameSurferEntry associated with this name, if one  
* exists. If the name does not appear in the database, this  
* method returns null.
```

```
*/
```

```
public NameSurferEntry findEntry(String name) {  
    // You need to turn this stub into a real implementation //  
    return null;  
}  
}
```



```
public class NameSurferDataBase implements NameSurferConstants {
```

```
/* Constructor: NameSurferDataBase(filename) */
```

```
/**
```

```
* Creates a new NameSurferDataBase and initializes it using the  
* data in the specified file. The constructor throws an error  
* exception if the requested file does not exist or if an error  
* occurs as the file is being read.
```

```
*/
```

```
public NameSurferDataBase(String filename) {  
    // You fill this in //  
}
```

name is case-insensitive


```
/* Method: findEntry(name) */
```

```
/**
```

```
* Returns the NameSurferEntry associated with this name, if one  
* exists. If the name does not appear in the database, this  
* method returns null.
```

```
*/
```

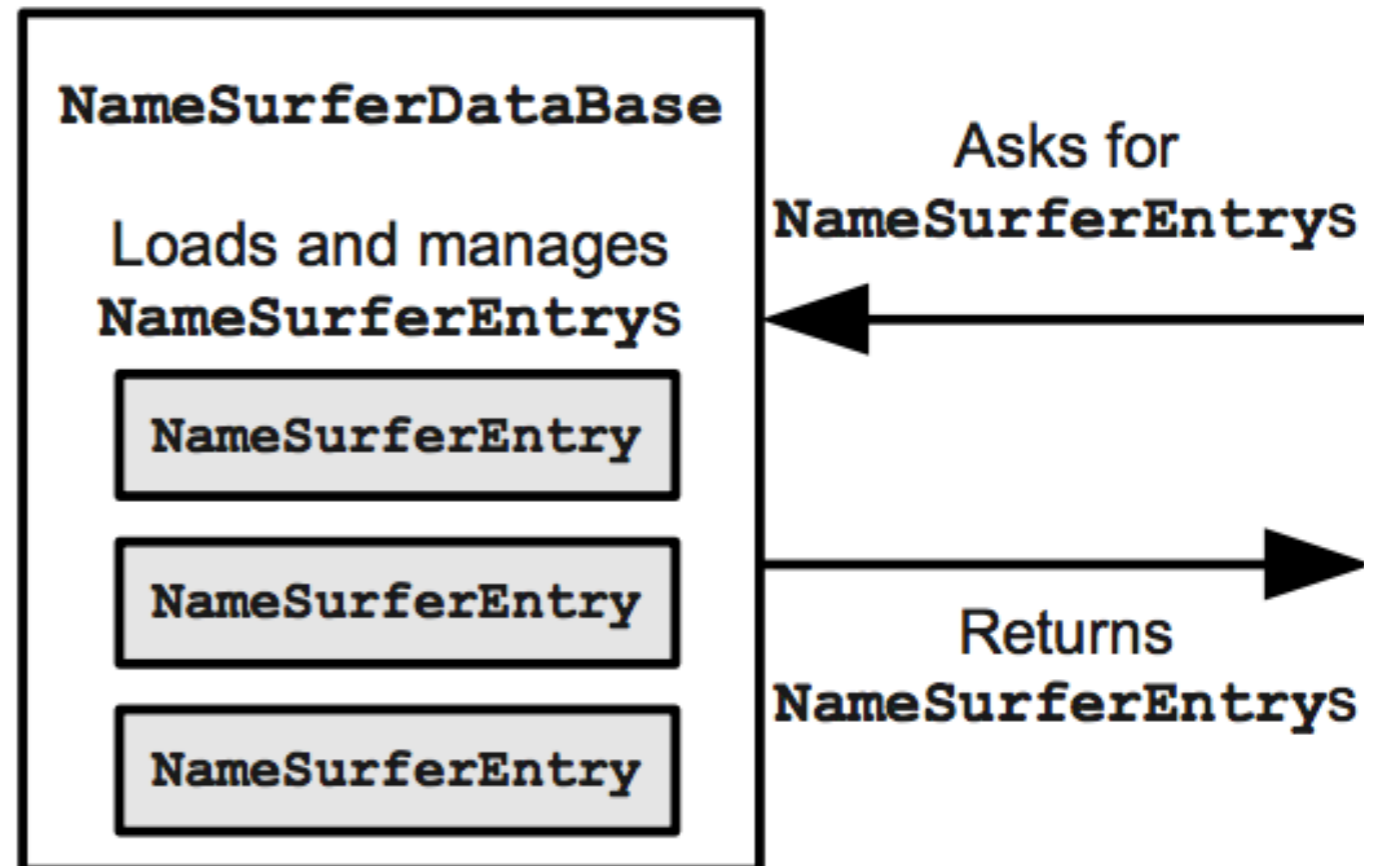
```
public NameSurferEntry findEntry(String name) {  
    // You need to turn this stub into a real implementation //  
    return null;  
}  
}
```



NameSurferDatabase.java

NamesData.txt

```
....  
Sam 58 69 99 131 168 236 278 380 467 408 466 997  
Samantha 0 0 0 0 0 0 272 107 26 5 7 63  
Samara 0 0 0 0 0 0 0 0 0 0 886 0  
Samir 0 0 0 0 0 0 0 0 920 0 798 0  
Sammie 537 545 351 325 333 396 565 772 930 0 0 0  
Sammy 0 887 544 299 202 262 321 395 575 639 755 0  
Samson 0 0 0 0 0 0 0 0 0 0 915 0  
Samuel 31 41 46 60 61 71 83 61 52 35 28 32  
Sandi 0 0 0 0 704 864 621 695 0 0 0 0  
Sandra 0 942 606 50 6 12 11 39 94 168 257 962  
....
```

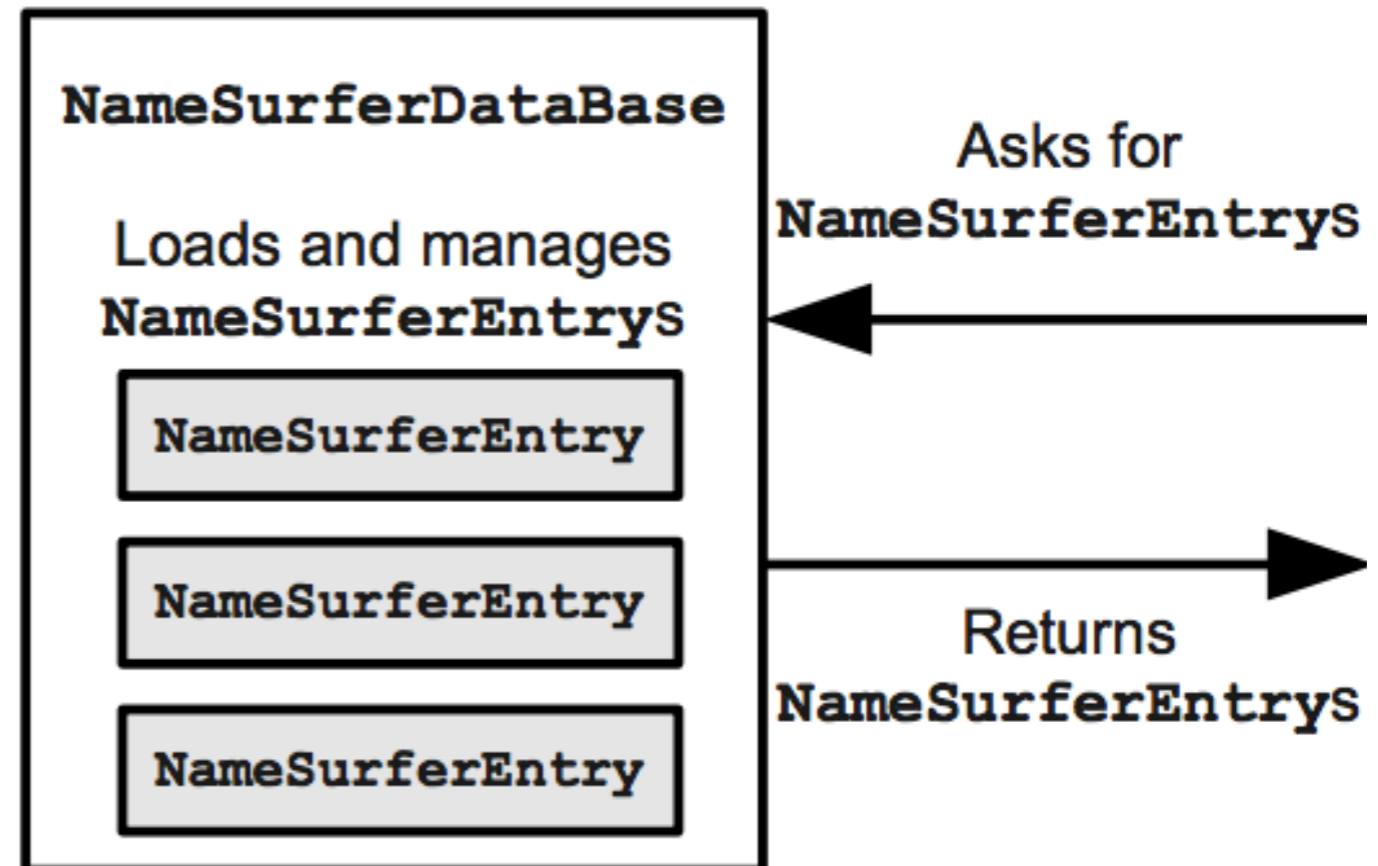


NameSurferDatabase.java

NamesData.txt

```
....  
Sam 58 69 99 131 168 236 278 380 467 408 466 997  
Samantha 0 0 0 0 0 0 272 107 26 5 7 63  
Samara 0 0 0 0 0 0 0 0 0 0 886 0  
Samir 0 0 0 0 0 0 0 920 0 798 0  
Sammie 537 545 351 325 333 396 565 772 930 0 0 0  
Sammy 0 887 544 299 202 262 321 395 575 639 755 0  
Samson 0 0 0 0 0 0 0 0 0 915 0  
Samuel 31 41 46 60 61 71 83 61 52 35 28 32  
Sandi 0 0 0 0 704 864 621 695 0 0 0  
Sandra 0 942 606 50 6 12 11 39 94 168 257 962  
....
```

rank 0 means the name did not appear in the top 1000 names for that year



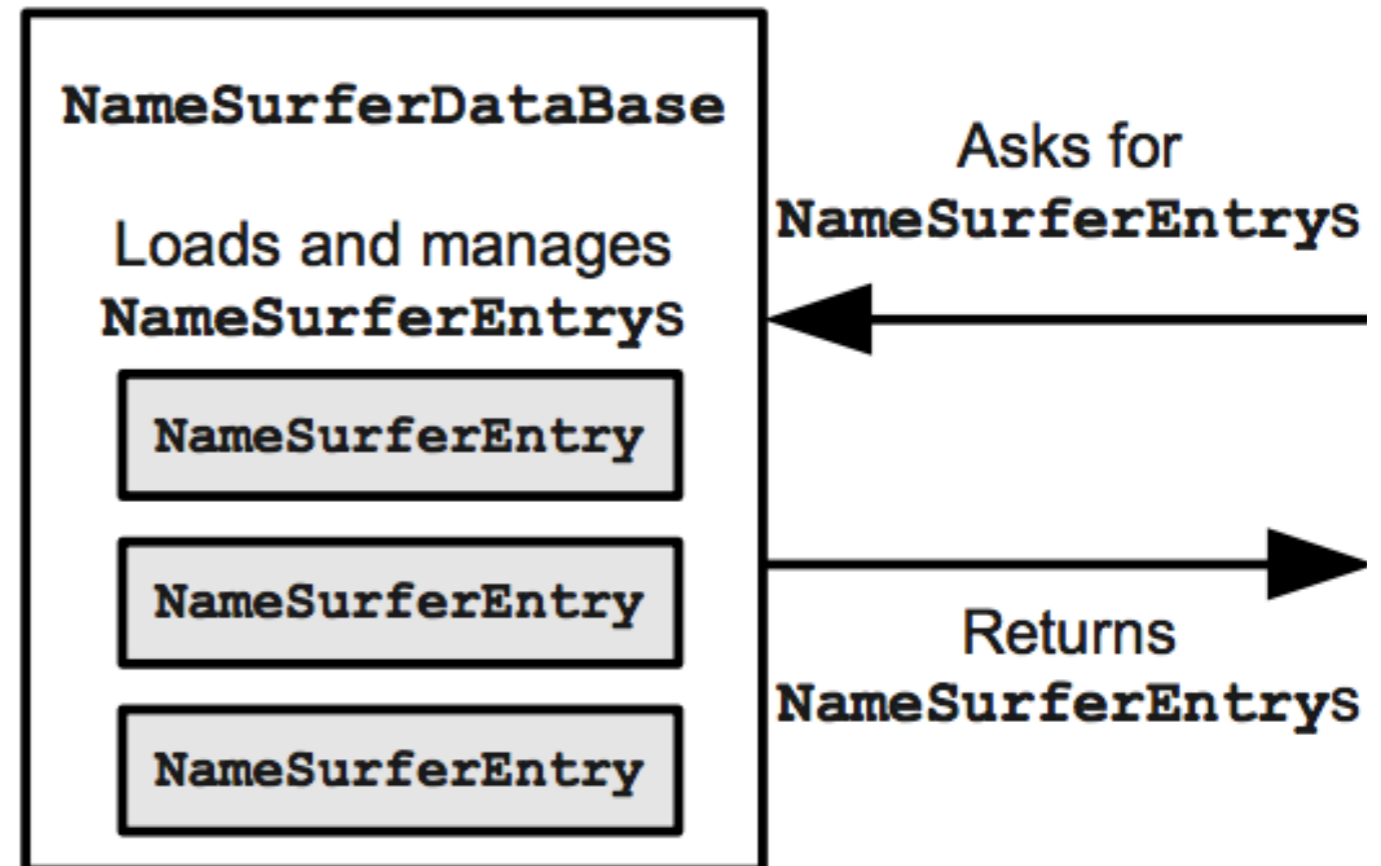
NameSurferDatabase.java

```
String line = rd.readLine();
```

NamesData.txt

```
....  
Sam 58 69 99 131 168 236 278 380 467 408 466 997  
Samantha 0 0 0 0 0 0 272 107 26 5 7 63  
Samara 0 0 0 0 0 0 0 0 0 0 886 0  
Samir 0 0 0 0 0 0 0 0 920 0 798 0  
Sammie 537 545 351 325 333 396 565 772 930 0 0 0  
Sammy 0 887 544 299 202 262 321 395 575 639 755 0  
Samson 0 0 0 0 0 0 0 0 0 0 915 0  
Samuel 31 41 46 60 61 71 83 61 52 35 28 32  
Sandi 0 0 0 0 704 864 621 695 0 0 0 0  
Sandra 0 942 606 50 6 12 11 39 94 168 257 962  
....
```

rank 0 means the name did not
appear in the top 1000 names for
that year



NameSurferDatabase.java

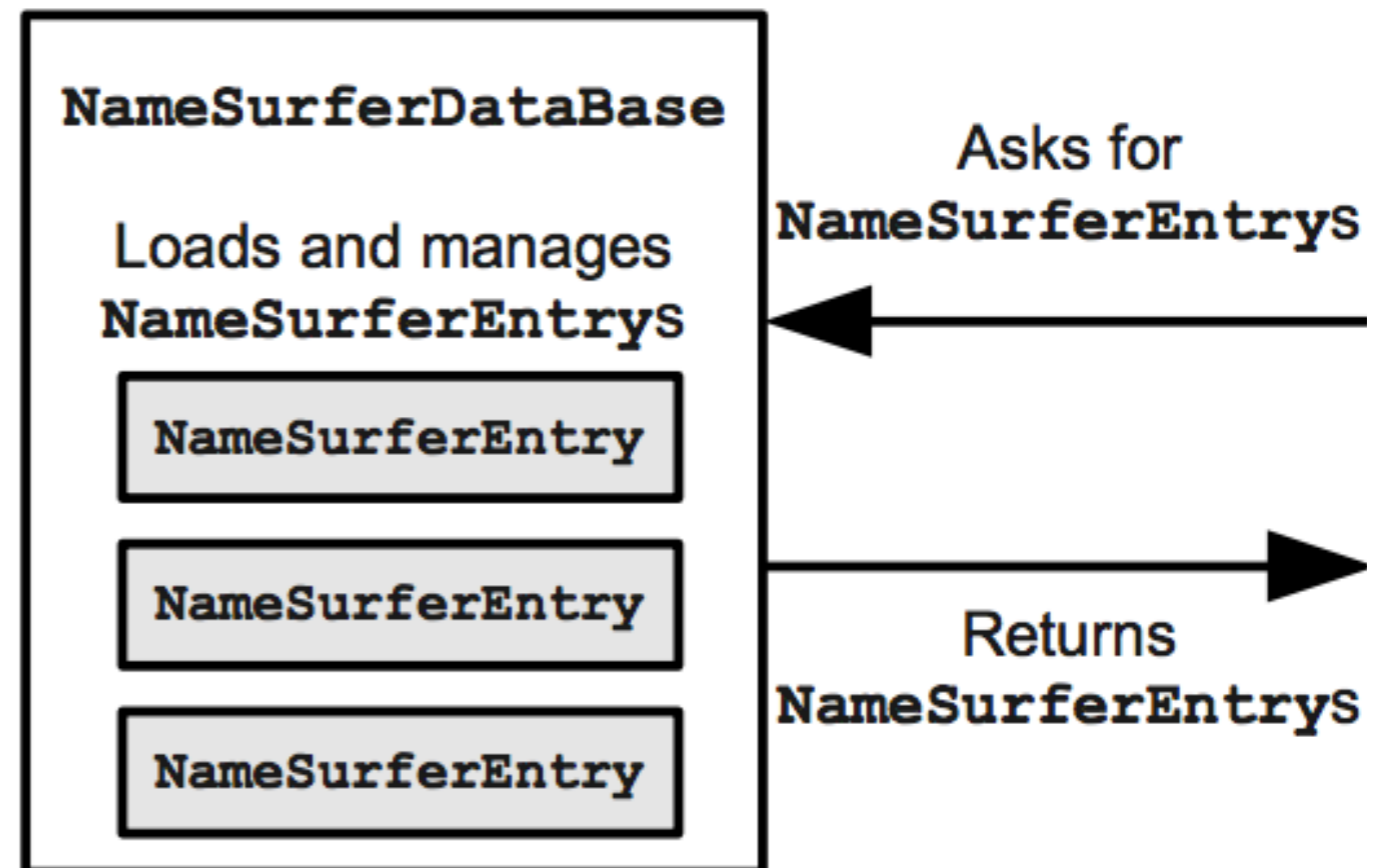
```
String line = rd.readLine();
```

```
NameSurferEntry entry = new NameSurferEntry(line);
```

NamesData.txt

```
...  
Sam 58 69 99 131 168 236 278 380 467 408 466 997  
Samantha 0 0 0 0 0 0 272 107 26 5 7 63  
Samara 0 0 0 0 0 0 0 0 0 0 886 0  
Samir 0 0 0 0 0 0 0 0 920 0 798 0  
Sammie 537 545 351 325 333 396 565 772 930 0 0 0  
Sammy 0 887 544 299 202 262 321 395 575 639 755 0  
Samson 0 0 0 0 0 0 0 0 0 0 915 0  
Samuel 31 41 46 60 61 71 83 61 52 35 28 32  
Sandi 0 0 0 0 704 864 621 695 0 0 0 0  
Sandra 0 942 606 50 6 12 11 39 94 168 257 962  
...
```

rank 0 means the name did not
appear in the top 1000 names for
that year



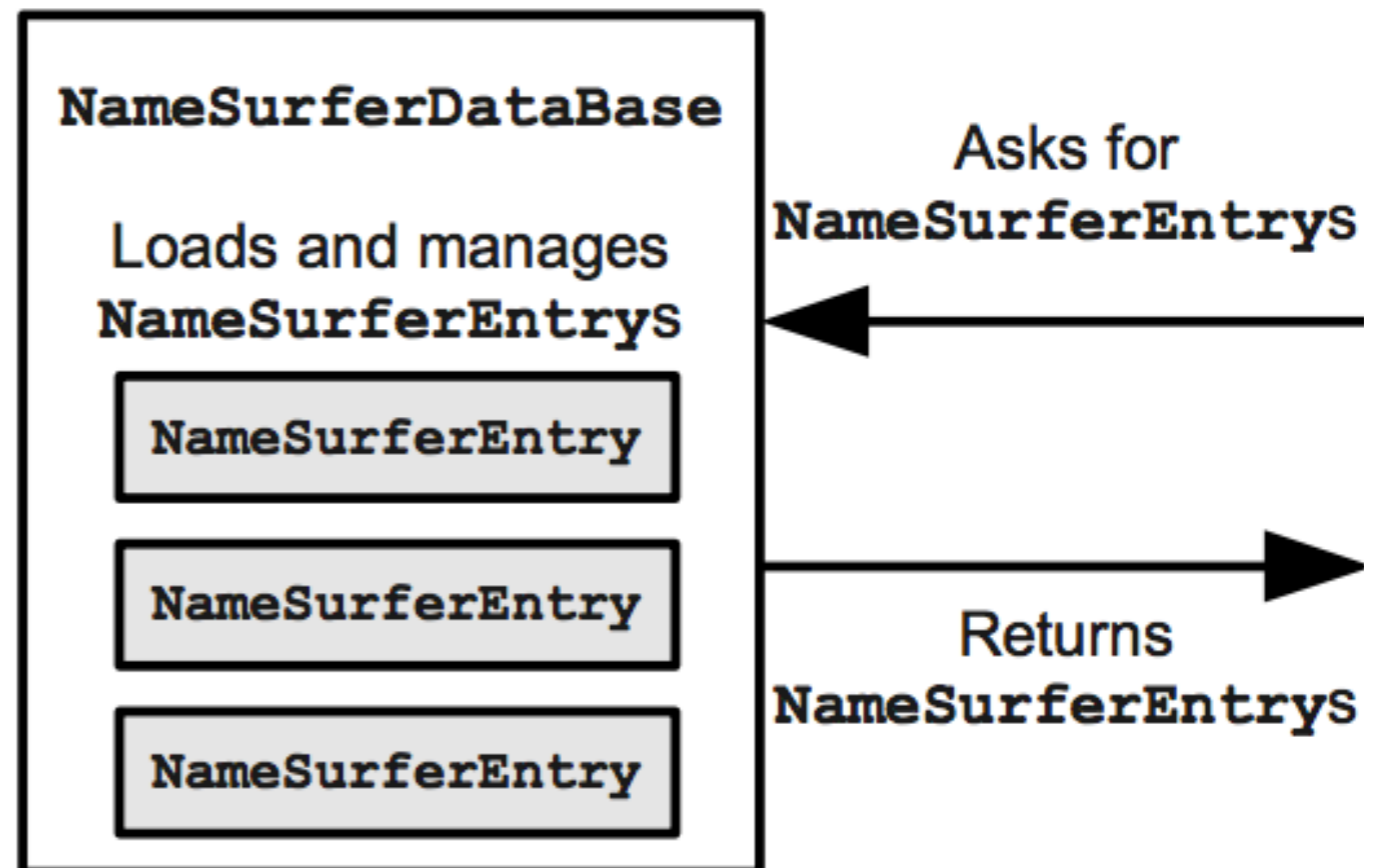
NameSurferDatabase.java

```
String line = rd.readLine();  
NameSurferEntry entry = new NameSurferEntry(line);  
//Store this NameSurferEntry so it can be retrieved
```

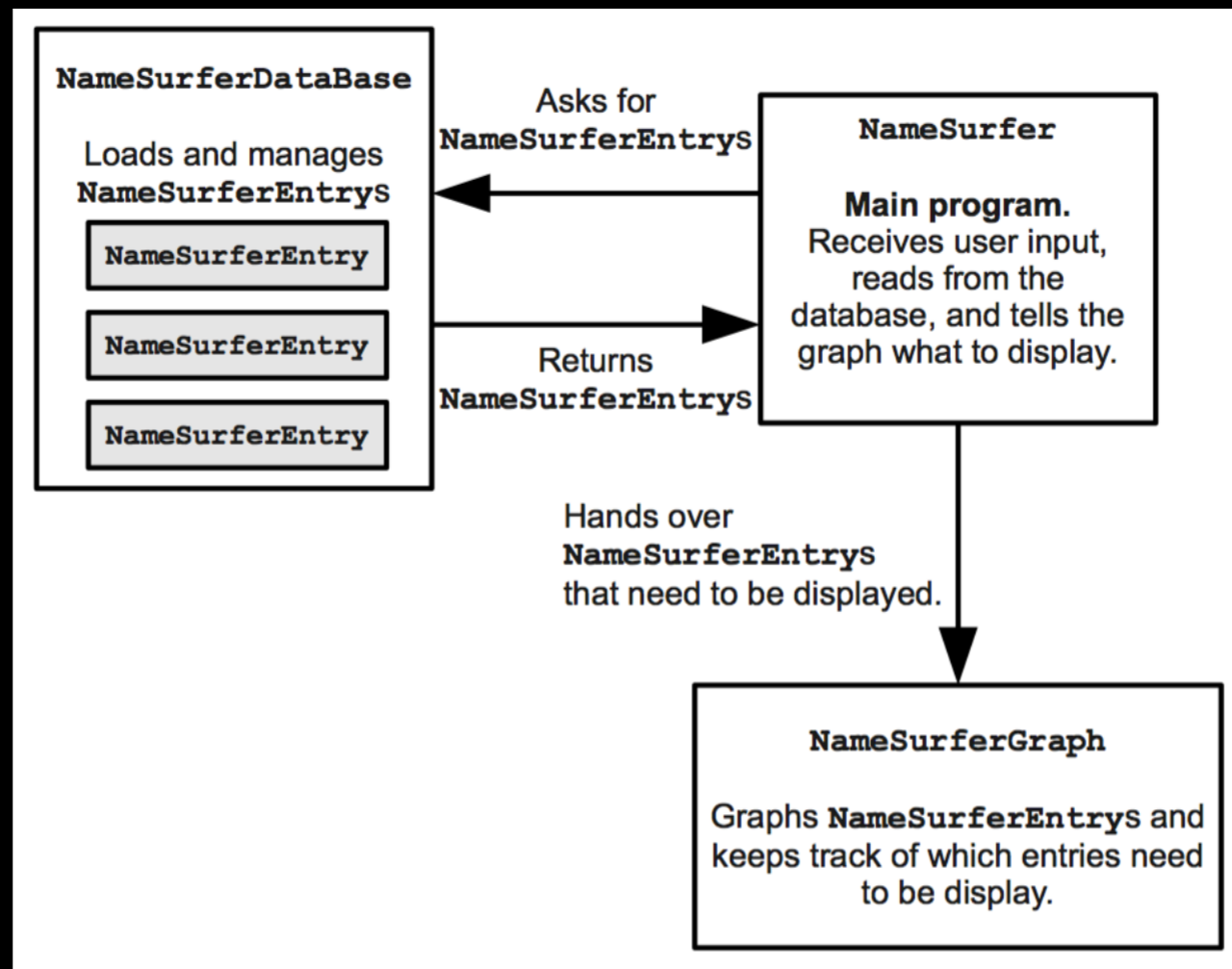
NamesData.txt

```
....  
Sam 58 69 99 131 168 236 278 380 467 408 466 997  
Samantha 0 0 0 0 0 0 272 107 26 5 7 63  
Samara 0 0 0 0 0 0 0 0 0 0 886 0  
Samir 0 0 0 0 0 0 0 920 0 798 0  
Sammie 537 545 351 325 333 396 565 772 930 0 0 0  
Sammy 0 887 544 299 202 262 321 395 575 639 755 0  
Samson 0 0 0 0 0 0 0 0 0 0 915 0  
Samuel 31 41 46 60 61 71 83 61 52 35 28 32  
Sandi 0 0 0 0 704 864 621 695 0 0 0 0  
Sandra 0 942 606 50 6 12 11 39 94 168 257 962  
....
```

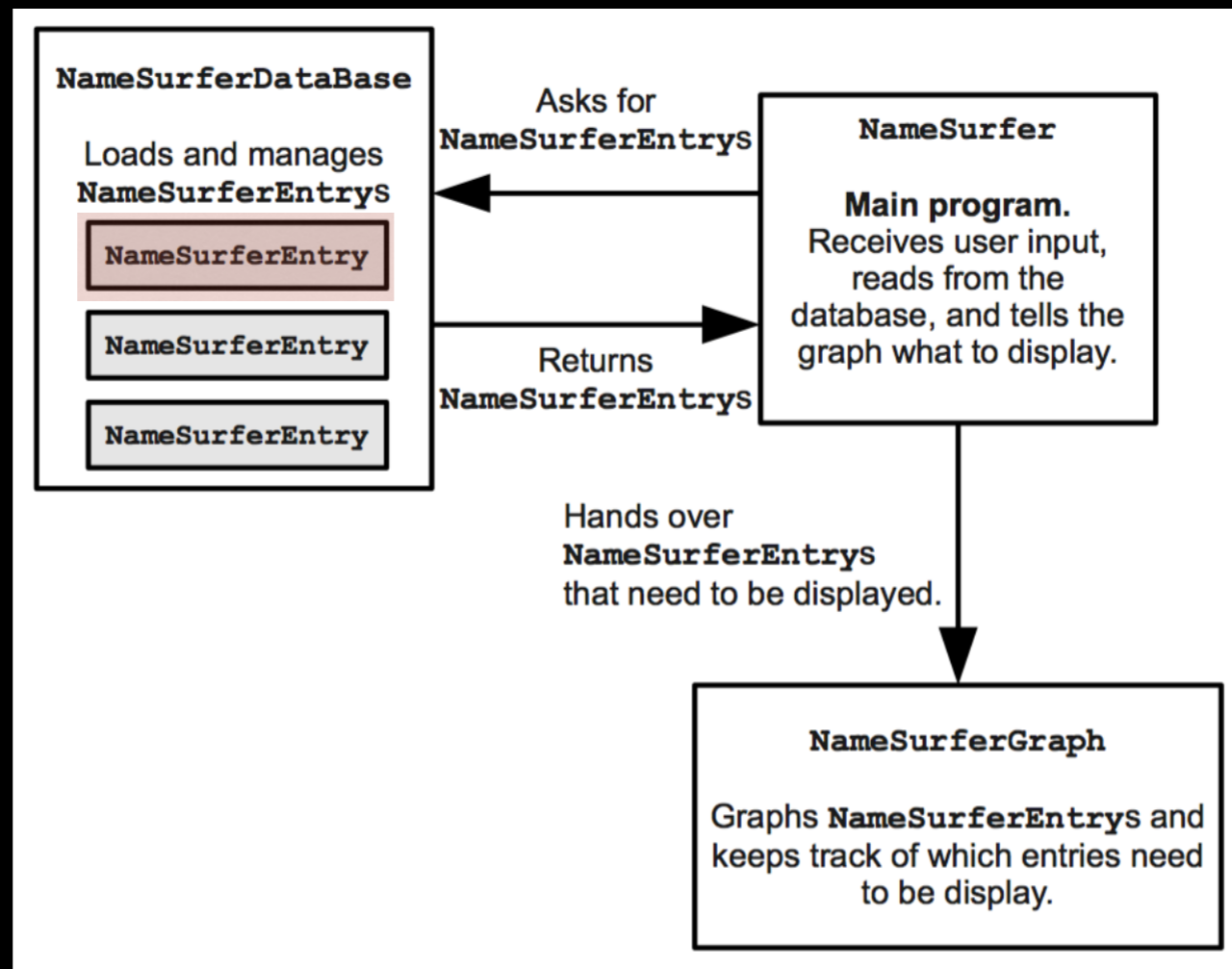
rank 0 means the name did not
appear in the top 1000 names for
that year



NameSurferEntry



NameSurferEntry



```

public class NameSurferEntry implements NameSurferConstants {

    /* Constructor: NameSurferEntry(line) */
    /**
     * Creates a new NameSurferEntry from a data line as it appears
     * in the data file. Each line begins with the name, which is
     * followed by integers giving the rank of that name for each
     * decade.
     */
    public NameSurferEntry(String line) {
        // You fill this in //
    }

    /* Method: getName() */
    /**
     * Returns the name associated with this entry.
     */
    public String getName() {
        // You need to turn this stub into a real implementation //
        return null;
    }

    /* Method: getRank(decade) */
    /**
     * Returns the rank associated with an entry for a particular
     * decade. The decade value is an integer indicating how many
     * decades have passed since the first year in the database,
     * which is given by the constant START_DECADE. If a name does
     * not appear in a decade, the rank value is 0.
     */
    public int getRank(int decade) {
        // You need to turn this stub into a real implementation //
        return 0;
    }

    /* Method: toString() */
    /**
     * Returns a string that makes it easy to see the value of a
     * NameSurferEntry.
     */
    public String toString() {
        // You need to turn this stub into a real implementation //
        return "";
    }
}

```

```
public class NameSurferEntry implements NameSurferConstants {

    /* Constructor: NameSurferEntry(line) */
    /**
     * Creates a new NameSurferEntry from a data line as it appears
     * in the data file. Each line begins with the name, which is
     * followed by integers giving the rank of that name for each
     * decade.
     */
    public NameSurferEntry(String line) {
        // You fill this in //
    }

    /* Method: getName() */
    /**
     * Returns the name associated with this entry.
     */
    public String getName() {
        // You need to turn this stub into a real implementation //
        return null;
    }

    /* Method: getRank(decade) */
    /**
     * Returns the rank associated with an entry for a particular
     * decade. The decade value is an integer indicating how many
     * decades have passed since the first year in the database,
     * which is given by the constant START_DECAD. If a name does
     * not appear in a decade, the rank value is 0.
     */
    public int getRank(int decade) {
        // You need to turn this stub into a real implementation //
        return 0;
    }

    /* Method: toString() */
    /**
     * Returns a string that makes it easy to see the value of a
     * NameSurferEntry.
     */
    public String toString() {
        // You need to turn this stub into a real implementation //
        return "";
    }
}
```



```
public class NameSurferEntry implements NameSurferConstants {

    /* Constructor: NameSurferEntry(line) */
    /**
     * Creates a new NameSurferEntry from a data line as it appears
     * in the data file. Each line begins with the name, which is
     * followed by integers giving the rank of that name for each
     * decade.
     */
    public NameSurferEntry(String line) {
        // You fill this in //
    }

    /* Method: getName() */
    /**
     * Returns the name associated with this entry.
     */
    public String getName() {
        // You need to turn this stub into a real implementation //
        return null;
    }

    /* Method: getRank(decade) */
    /**
     * Returns the rank associated with an entry for a particular
     * decade. The decade value is an integer indicating how many
     * decades have passed since the first year in the database,
     * which is given by the constant START_DECAD. If a name does
     * not appear in a decade, the rank value is 0.
     */
    public int getRank(int decade) {
        // You need to turn this stub into a real implementation //
        return 0;
    }

    /* Method: toString() */
    /**
     * Returns a string that makes it easy to see the value of a
     * NameSurferEntry.
     */
    public String toString() {
        // You need to turn this stub into a real implementation //
        return "";
    }
}
```




```
public class NameSurferEntry implements NameSurferConstants {

    /* Constructor: NameSurferEntry(line) */
    /**
     * Creates a new NameSurferEntry from a data line as it appears
     * in the data file. Each line begins with the name, which is
     * followed by integers giving the rank of that name for each
     * decade.
     */
    public NameSurferEntry(String line) {
        // You fill this in //
    }

    /* Method: getName() */
    /**
     * Returns the name associated with this entry.
     */
    public String getName() {
        // You need to turn this stub into a real implementation //
        return null;
    }

    /* Method: getRank(decade) */
    /**
     * Returns the rank associated with an entry for a particular
     * decade. The decade value is an integer indicating how many
     * decades have passed since the first year in the database,
     * which is given by the constant START_DECAD. If a name does
     * not appear in a decade, the rank value is 0.
     */
    public int getRank(int decade) {
        // You need to turn this stub into a real implementation //
        return 0;
    }

    /* Method: toString() */
    /**
     * Returns a string that makes it easy to see the value of a
     * NameSurferEntry.
     */
    public String toString() {
        // You need to turn this stub into a real implementation //
        return "";
    }
}
```



```

public class NameSurferEntry implements NameSurferConstants {

    /* Constructor: NameSurferEntry(line) */
    /**
     * Creates a new NameSurferEntry from a data line as it appears
     * in the data file. Each line begins with the name, which is
     * followed by integers giving the rank of that name for each
     * decade.
     */
    public NameSurferEntry(String line) {
        // You fill this in //
    }

    /* Method: getName() */
    /**
     * Returns the name associated with this entry.
     */
    public String getName() {
        // You need to turn this stub into a real implementation //
        return null;
    }

    /* Method: getRank(decade) */
    /**
     * Returns the rank associated with an entry for a particular
     * decade. The decade value is an integer indicating how many
     * decades have passed since the first year in the database,
     * which is given by the constant START_DECAD. If a name does
     * not appear in a decade, the rank value is 0.
     */
    public int getRank(int decade) {
        // You need to turn this stub into a real implementation //
        return 0;
    }

    /* Method: toString() */
    /**
     * Returns a string that makes it easy to see the value of a
     * NameSurferEntry.
     */
    public String toString() {
        // You need to turn this stub into a real implementation //
        return "";
    }
}

```



}

NameSurferEntry.java

```
String getName()  
int getRank(int decade)  
String toString()
```

NameSurferEntry.java

NamesData.txt

```
...
Sam 58 69 99 131 168 236 278 380 467 408 466 997
Samantha 0 0 0 0 0 0 272 107 26 5 7 63
Samara 0 0 0 0 0 0 0 0 0 886 0
Samir 0 0 0 0 0 0 0 920 0 798 0
Sammie 537 545 351 325 333 396 565 772 930 0 0 0
Sammy 0 887 544 299 202 262 321 395 575 639 755 0
Samson 0 0 0 0 0 0 0 0 0 915 0
Samuel 31 41 46 60 61 71 83 61 52 35 28 32
Sandi 0 0 0 0 704 864 621 695 0 0 0 0
Sandra 0 942 606 50 6 12 11 39 94 168 257 962
...
```

```
String getName()
int getRank(int decade)
String toString()
```

NameSurferEntry.java

NamesData.txt

```
Sam 58 69 99 131 168 236 278 380 467 408 466 997
Samantha 0 0 0 0 0 0 272 107 26 5 7 63
Samara 0 0 0 0 0 0 0 0 0 0 886 0
Samir 0 0 0 0 0 0 0 0 920 0 798 0
Sammie 537 545 351 325 333 396 565 772 930 0 0 0
Sammy 0 887 544 299 202 262 321 395 575 639 755 0
Samson 0 0 0 0 0 0 0 0 0 0 915 0
Samuel 31 41 46 60 61 71 83 61 52 35 28 32
Sandi 0 0 0 0 704 864 621 695 0 0 0 0
Sandra 0 942 606 50 6 12 11 39 94 168 257 962
```

```
String getName()
int getRank(int decade)
String toString()
```

NameSurferEntry.java

"Sam 58 69 99 131 168 236 278 380 467 408 466 997"

```
String getName()  
int getRank(int decade)  
String toString()
```

NameSurferEntry.java

`"Sam 58 69 99 131 168 236 278 380 467 408 466 997"`

rankData

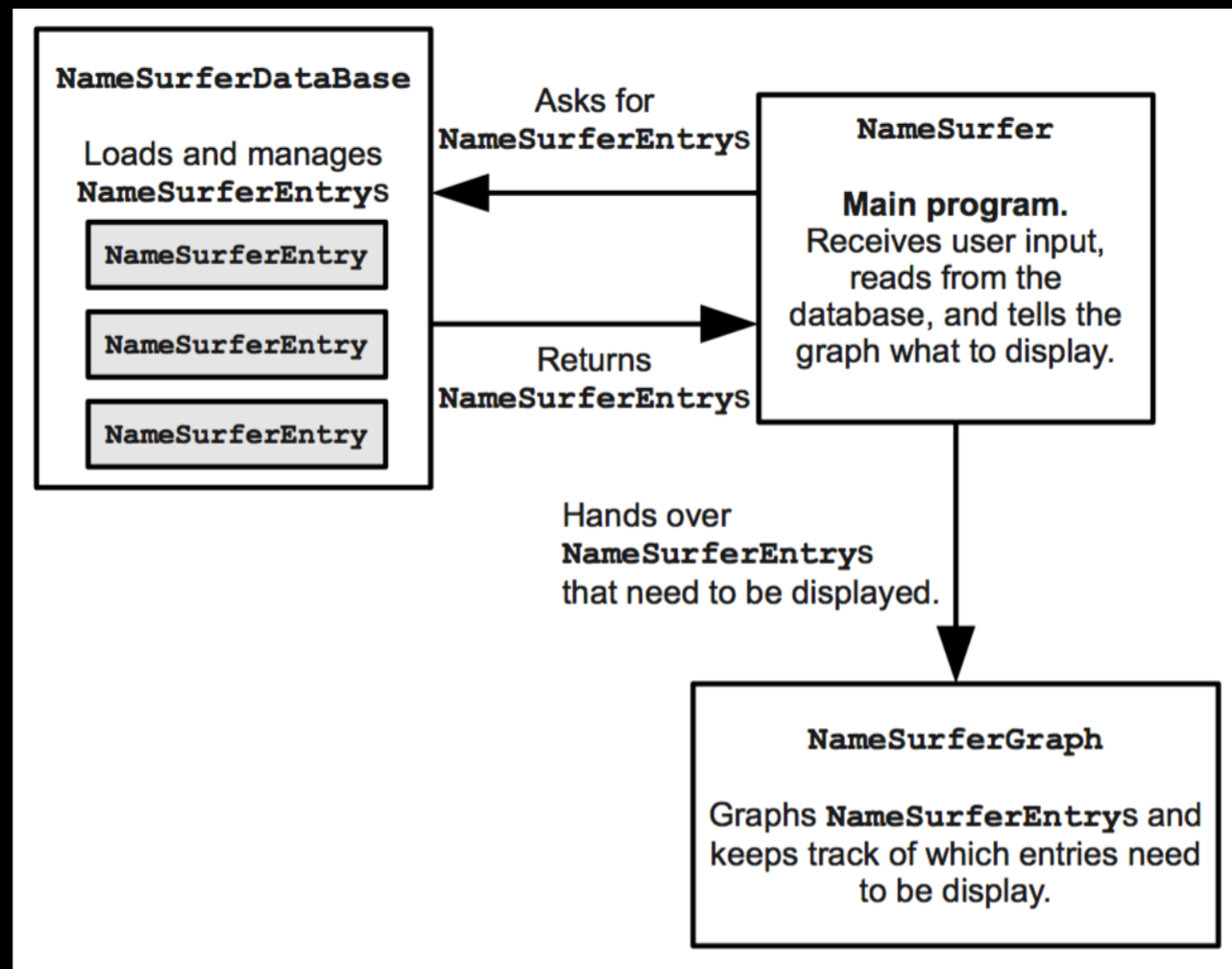
```
58 69 99 131 168 236 278 380 467 408 466 997
```

name

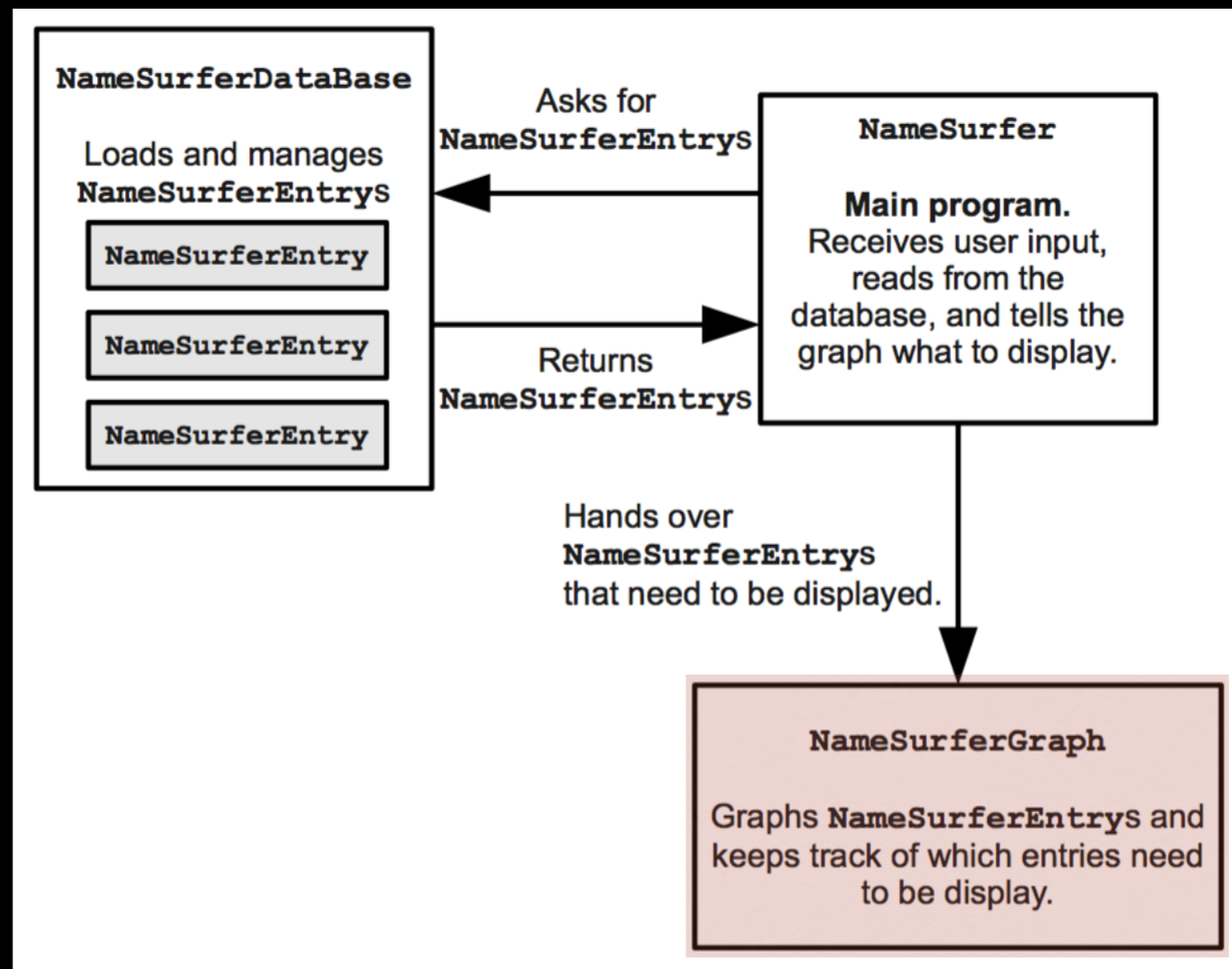
```
"Sam"
```

```
String getName()  
int getRank(int decade)  
String toString()
```

NameSurferGraph



NameSurferGraph



```

public class NameSurferGraph extends GCanvas
    implements NameSurferConstants, ComponentListener {
    /**
     * Creates a new NameSurferGraph object that displays the data.
     */
    public NameSurferGraph() {
        addComponentListener(this);
        // You fill in the rest //
    }

    /**
     * Clears the list of name surfer entries stored inside this class.
     */
    public void clear() {
        // You fill this in //
    }

    /* Method: addEntry(entry) */
    /**
     * Adds a new NameSurferEntry to the list of entries on the display.
     * Note that this method does not actually draw the graph, but
     * simply stores the entry; the graph is drawn by calling update.
     */
    public void addEntry(NameSurferEntry entry) {
        // You fill this in //
    }

    /**
     * Updates the display image by deleting all the graphical objects
     * from the canvas and then reassembling the display according to
     * the list of entries. Your application must call update after
     * calling either clear or addEntry; update is also called whenever
     * the size of the canvas changes.
     */
    public void update() {
        // You fill this in //
    }

    /* Implementation of the ComponentListener interface */
    public void componentHidden(ComponentEvent e) { }
    public void componentMoved(ComponentEvent e) { }
    public void componentResized(ComponentEvent e) { update(); }
    public void componentShown(ComponentEvent e) { }
}

```

```
public class NameSurferGraph extends GCanvas
    implements NameSurferConstants, ComponentListener {
    /**
     * Creates a new NameSurferGraph object that displays the data.
     */
    public NameSurferGraph() {
        addComponentListener(this);
        // You fill in the rest //
    }

    /**
     * Clears the list of name surfer entries stored inside this class.
     */
    public void clear() {
        // You fill this in //
    }

    /* Method: addEntry(entry) */
    /**
     * Adds a new NameSurferEntry to the list of entries on the display.
     * Note that this method does not actually draw the graph, but
     * simply stores the entry; the graph is drawn by calling update.
     */
    public void addEntry(NameSurferEntry entry) {
        // You fill this in //
    }

    /**
     * Updates the display image by deleting all the graphical objects
     * from the canvas and then reassembling the display according to
     * the list of entries. Your application must call update after
     * calling either clear or addEntry; update is also called whenever
     * the size of the canvas changes.
     */
    public void update() {
        // You fill this in //
    }

    /* Implementation of the ComponentListener interface */
    public void componentHidden(ComponentEvent e) { }
    public void componentMoved(ComponentEvent e) { }
    public void componentResized(ComponentEvent e) { update(); }
    public void componentShown(ComponentEvent e) { }
}
}
```



```
public class NameSurferGraph extends GCanvas
    implements NameSurferConstants, ComponentListener {
    /**
     * Creates a new NameSurferGraph object that displays the data.
     */
    public NameSurferGraph() {
        addComponentListener(this);
        // You fill in the rest //
    }

    /**
     * Clears the list of name surfer entries stored inside this class.
     */
    public void clear() {
        // You fill this in //
    }

    /* Method: addEntry(entry) */
    /**
     * Adds a new NameSurferEntry to the list of entries on the display.
     * Note that this method does not actually draw the graph, but
     * simply stores the entry; the graph is drawn by calling update.
     */
    public void addEntry(NameSurferEntry entry) {
        // You fill this in //
    }

    /**
     * Updates the display image by deleting all the graphical objects
     * from the canvas and then reassembling the display according to
     * the list of entries. Your application must call update after
     * calling either clear or addEntry; update is also called whenever
     * the size of the canvas changes.
     */
    public void update() {
        // You fill this in //
    }

    /* Implementation of the ComponentListener interface */
    public void componentHidden(ComponentEvent e) { }
    public void componentMoved(ComponentEvent e) { }
    public void componentResized(ComponentEvent e) { update(); }
    public void componentShown(ComponentEvent e) { }
}
}
```



```

public class NameSurferGraph extends GCanvas
    implements NameSurferConstants, ComponentListener {
    /**
     * Creates a new NameSurferGraph object that displays the data.
     */
    public NameSurferGraph() {
        addComponentListener(this);
        // You fill in the rest //
    }

    /**
     * Clears the list of name surfer entries stored inside this class.
     */
    public void clear() {
        // You fill this in //
    }

    /* Method: addEntry(entry) */
    /**
     * Adds a new NameSurferEntry to the list of entries on the display.
     * Note that this method does not actually draw the graph, but
     * simply stores the entry; the graph is drawn by calling update.
     */
    public void addEntry(NameSurferEntry entry) {
        // You fill this in //
    }

    /**
     * Updates the display image by deleting all the graphical objects
     * from the canvas and then reassembling the display according to
     * the list of entries. Your application must call update after
     * calling either clear or addEntry; update is also called whenever
     * the size of the canvas changes.
     */
    public void update() {
        // You fill this in //
    }

    /* Implementation of the ComponentListener interface */
    public void componentHidden(ComponentEvent e) { }
    public void componentMoved(ComponentEvent e) { }
    public void componentResized(ComponentEvent e) { update(); }
    public void componentShown(ComponentEvent e) { }
}

```



```

public class NameSurferGraph extends GCanvas
    implements NameSurferConstants, ComponentListener {
    /**
     * Creates a new NameSurferGraph object that displays the data.
     */
    public NameSurferGraph() {
        addComponentListener(this);
        // You fill in the rest //
    }

    /**
     * Clears the list of name surfer entries stored inside this class.
     */
    public void clear() {
        // You fill this in //
    }

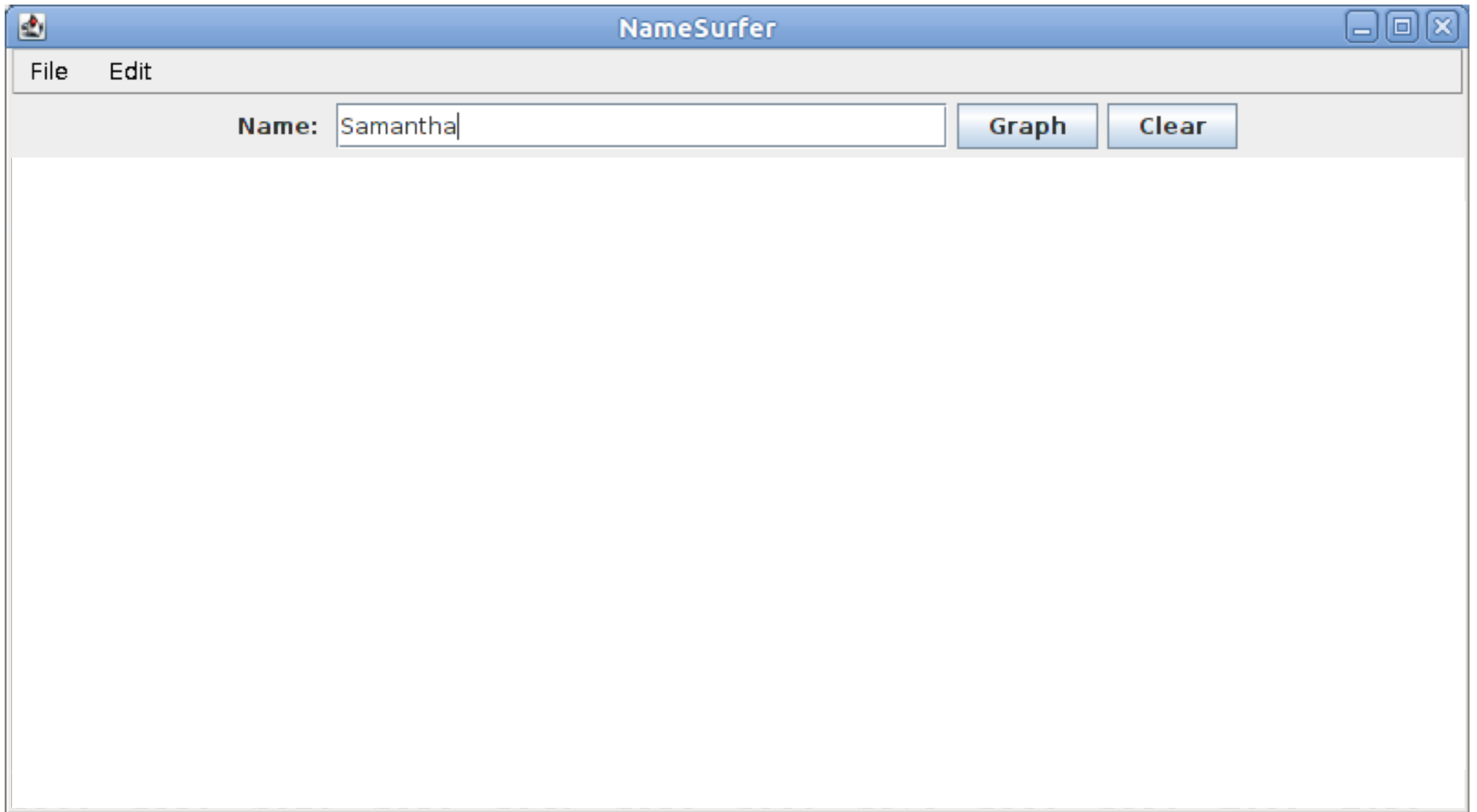
    /* Method: addEntry(entry) */
    /**
     * Adds a new NameSurferEntry to the list of entries on the display.
     * Note that this method does not actually draw the graph, but
     * simply stores the entry; the graph is drawn by calling update.
     */
    public void addEntry(NameSurferEntry entry) {
        // You fill this in //
    }

    /**
     * Updates the display image by deleting all the graphical objects
     * from the canvas and then reassembling the display according to
     * the list of entries. Your application must call update after
     * calling either clear or addEntry; update is also called whenever
     * the size of the canvas changes.
     */
    public void update() {
        // You fill this in //
    }

    /* Implementation of the ComponentListener interface */
    public void componentHidden(ComponentEvent e) { }
    public void componentMoved(ComponentEvent e) { }
    public void componentResized(ComponentEvent e) { update(); }
    public void componentShown(ComponentEvent e) { }
    }

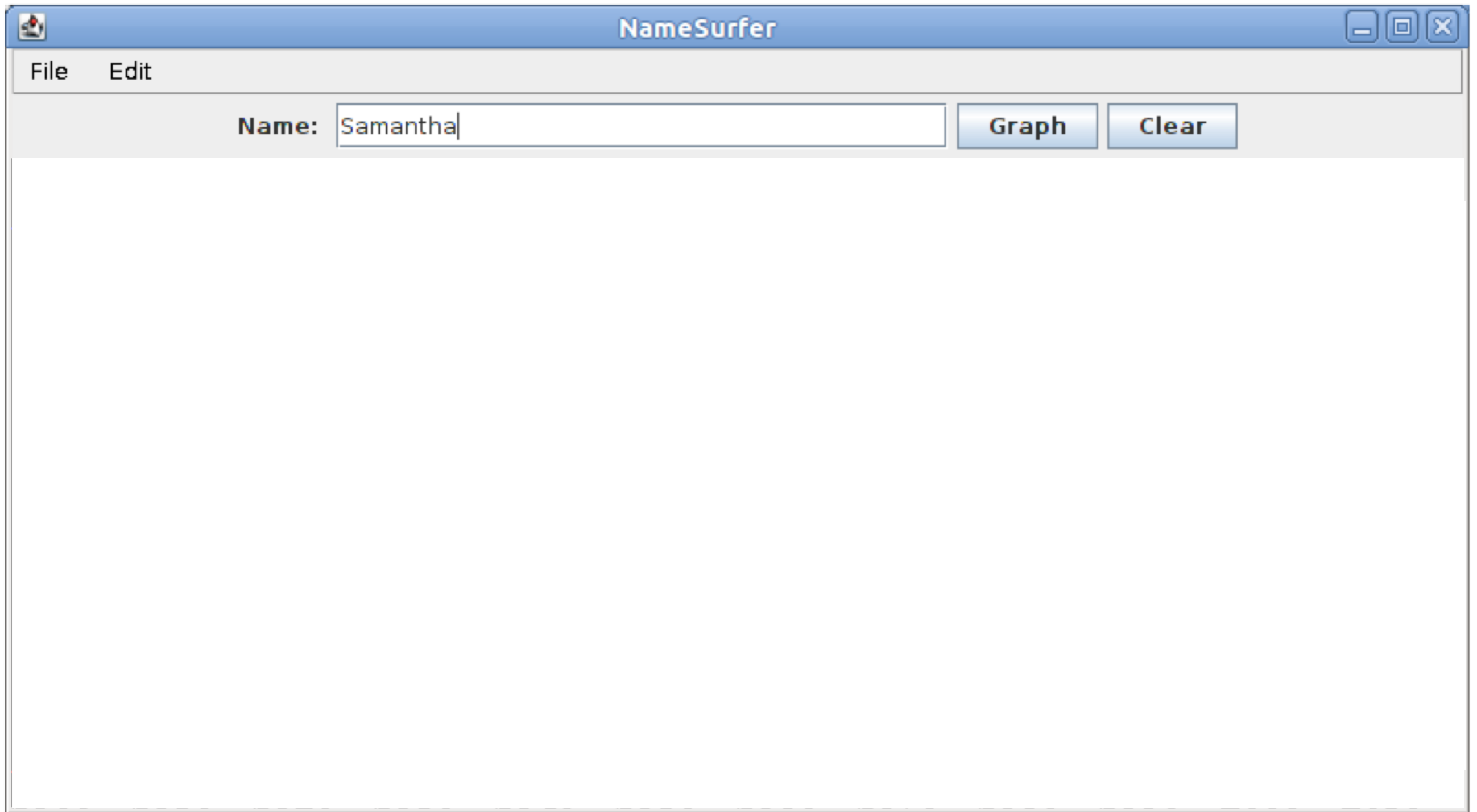
```





`entries`

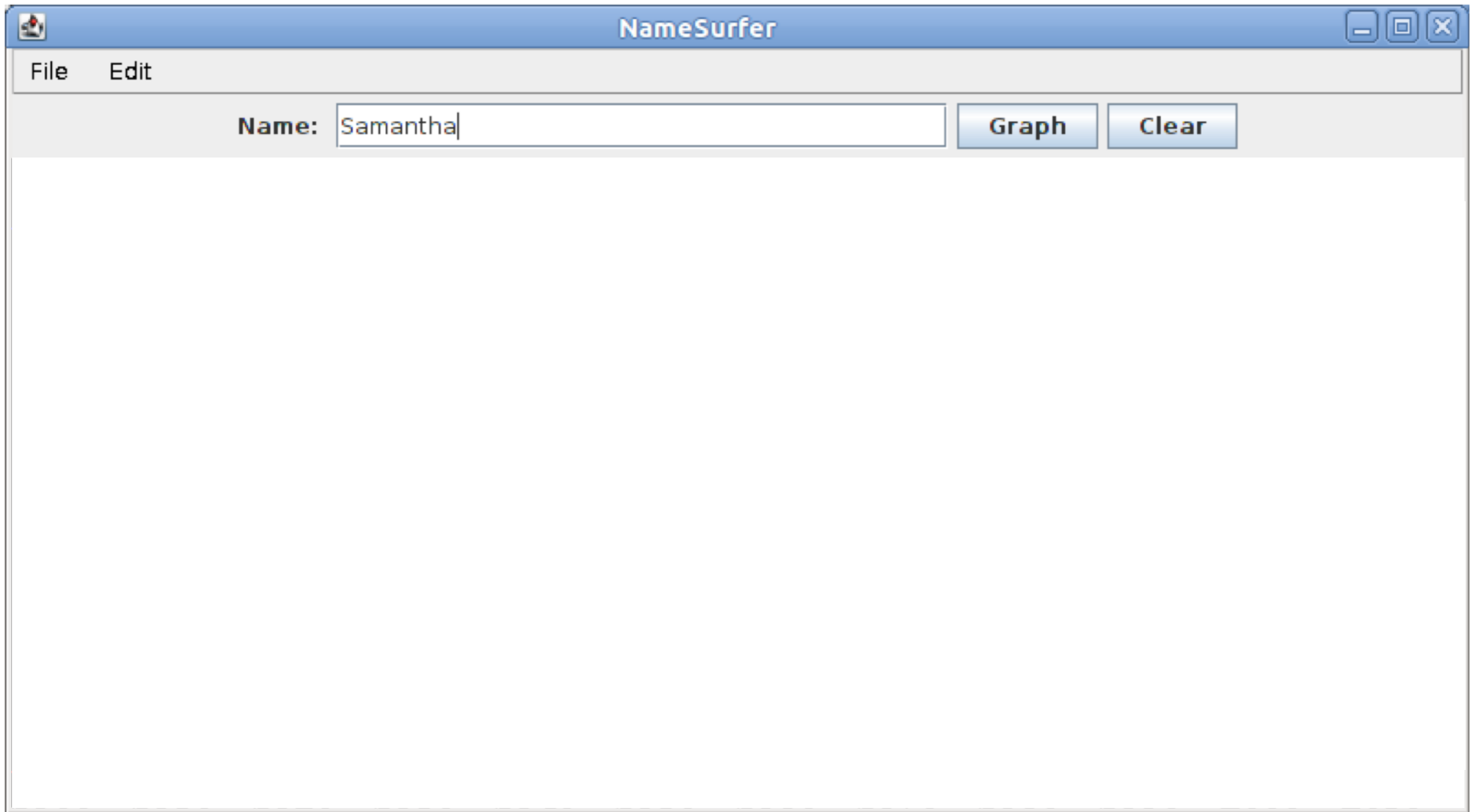
Sam NameSurferEntry



`entries`

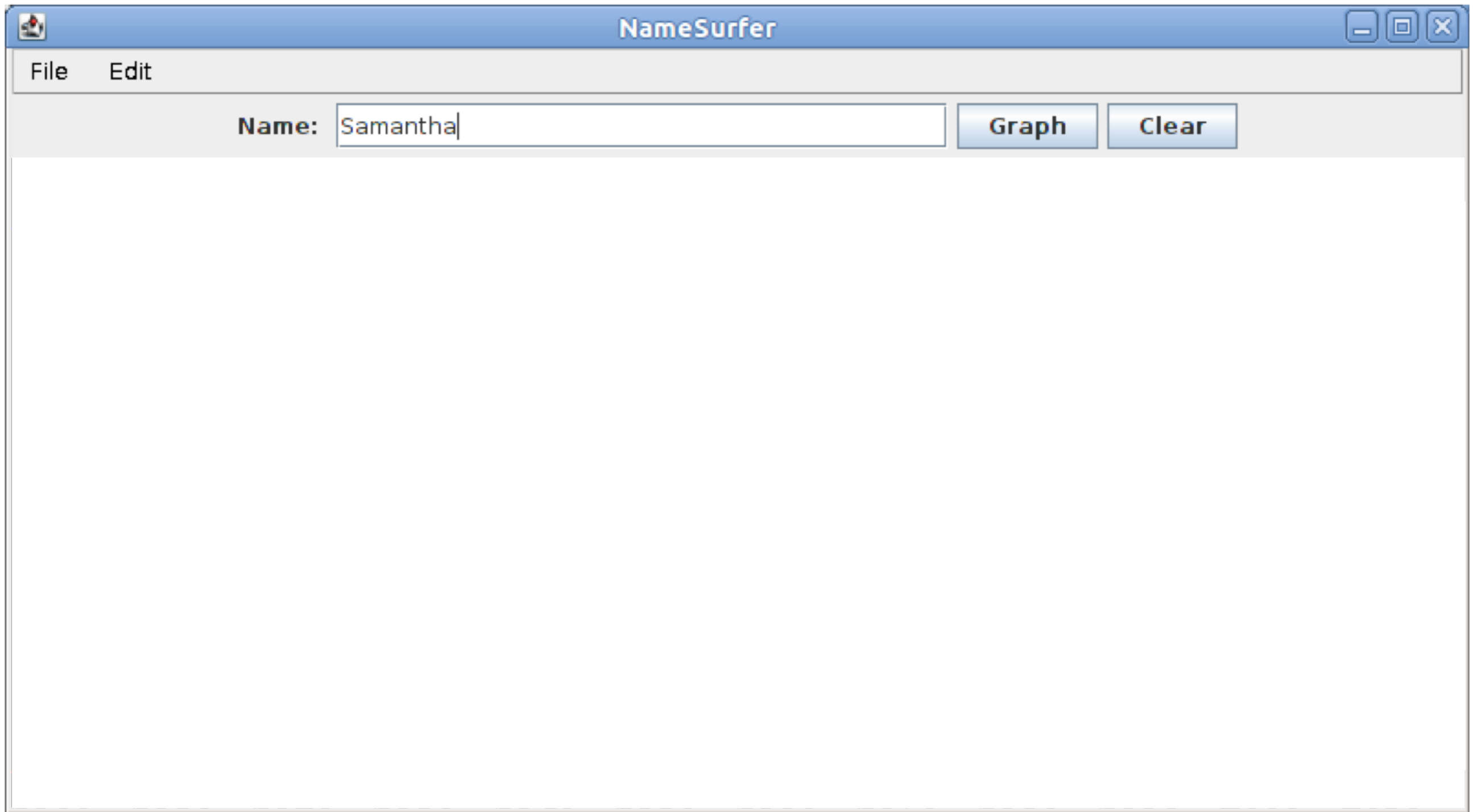
Sam NameSurferEntry

Samantha NameSurferEntry



entries

Sam NameSurferEntry (58 69 99 131 168 236 278 380 467 408 466 997)
Samantha NameSurferEntry (0 0 0 0 0 0 272 107 26 5 7 63)



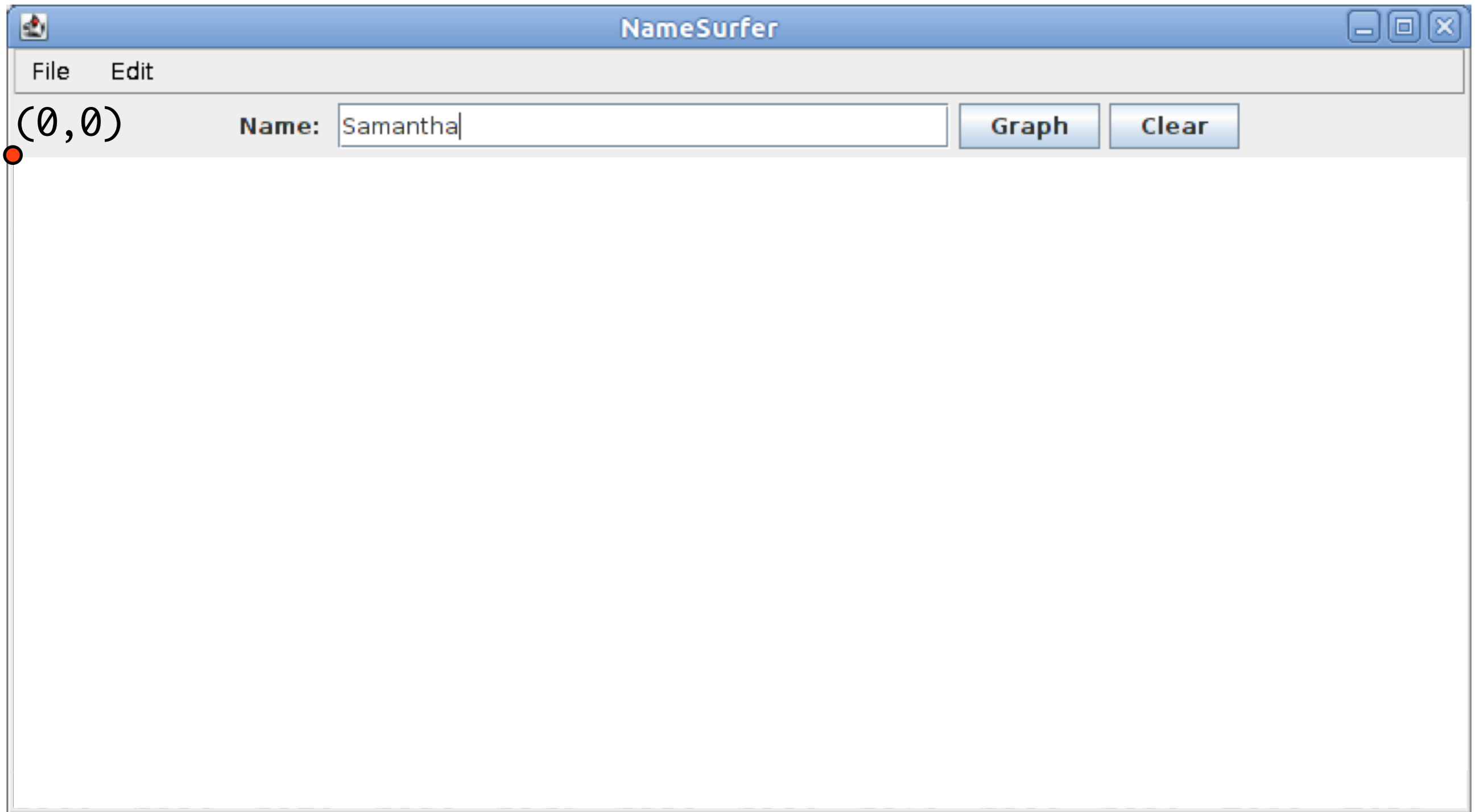
entries

Sam NameSurferEntry
(58 69 99 131 168 236 278 380 467 408 466 997)

Samantha NameSurferEntry
(0 0 0 0 0 272 107 26 5 7 63)

from NameSurferConstants

```
/** The first decade in the database */  
public static final int START_DECADE = 1900;  
  
/** The number of decades */  
public static final int NDECADES = 12;  
  
/** The maximum rank in the database */  
public static final int MAX_RANK = 1000;  
  
/** The number of pixels to reserve at the top and bottom */  
public static final int GRAPH_MARGIN_SIZE = 20;
```



entries

Sam NameSurferEntry
(58 69 99 131 168 236 278 380 467 408 466 997)

Samantha NameSurferEntry
(0 0 0 0 0 272 107 26 5 7 63)

from NameSurferConstants

```
/** The first decade in the database */  
public static final int START_DECADE = 1900;  
  
/** The number of decades */  
public static final int NDECADES = 12;  
  
/** The maximum rank in the database */  
public static final int MAX_RANK = 1000;  
  
/** The number of pixels to reserve at the top and bottom */  
public static final int GRAPH_MARGIN_SIZE = 20;
```



entries

Sam NameSurferEntry
(58 69 99 131 168 236 278 380 467 408 466 997)

Samantha NameSurferEntry
(0 0 0 0 0 272 107 26 5 7 63)

from NameSurferConstants

```
/** The first decade in the database */  
public static final int START_DECADE = 1900;  
  
/** The number of decades */  
public static final int NDECADES = 12;  
  
/** The maximum rank in the database */  
public static final int MAX_RANK = 1000;  
  
/** The number of pixels to reserve at the top and bottom */  
public static final int GRAPH_MARGIN_SIZE = 20;
```



entries

<p>Sam NameSurferEntry (58 69 99 131 168 236 278 380 467 408 466 997)</p>
<p>Samantha NameSurferEntry (0 0 0 0 0 272 107 26 5 7 63)</p>

from NameSurferConstants

```

/** The first decade in the database */
public static final int START_DECADE = 1900;

/** The number of decades */
public static final int NDECADES = 12;

/** The maximum rank in the database */
public static final int MAX_RANK = 1000;

/** The number of pixels to reserve at the top and bottom */
public static final int GRAPH_MARGIN_SIZE = 20;

```



entries

<p>Sam NameSurferEntry (58 69 99 131 168 236 278 380 467 408 466 997)</p>
<p>Samantha NameSurferEntry (0 0 0 0 0 272 107 26 5 7 63)</p>

from NameSurferConstants

```

/** The first decade in the database */
public static final int START_DECADE = 1900;

/** The number of decades */
public static final int NDECADES = 12;

/** The maximum rank in the database */
public static final int MAX_RANK = 1000;

/** The number of pixels to reserve at the top and bottom */
public static final int GRAPH_MARGIN_SIZE = 20;

```



entries

<p>Sam NameSurferEntry (58 69 99 131 168 236 278 380 467 408 466 997)</p>
<p>Samantha NameSurferEntry (0 0 0 0 0 272 107 26 5 7 63)</p>

from NameSurferConstants

```

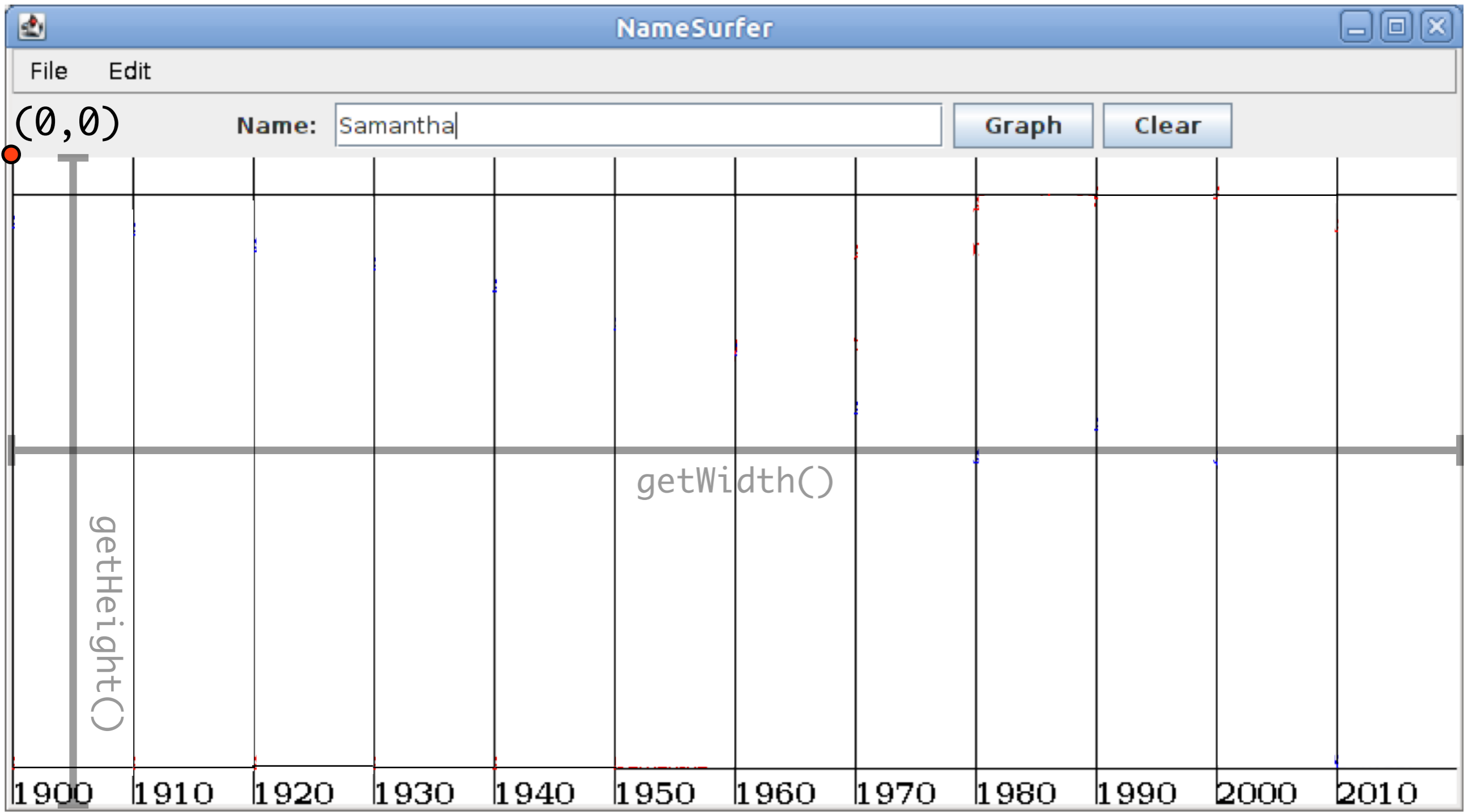
/** The first decade in the database */
public static final int START_DECADE = 1900;

/** The number of decades */
public static final int NDECADES = 12;

/** The maximum rank in the database */
public static final int MAX_RANK = 1000;

/** The number of pixels to reserve at the top and bottom */
public static final int GRAPH_MARGIN_SIZE = 20;

```



entries

Sam NameSurferEntry (58 69 99 131 168 236 278 380 467 408 466 997)
Samantha NameSurferEntry (0 0 0 0 0 272 107 26 5 7 63)

from NameSurferConstants

```

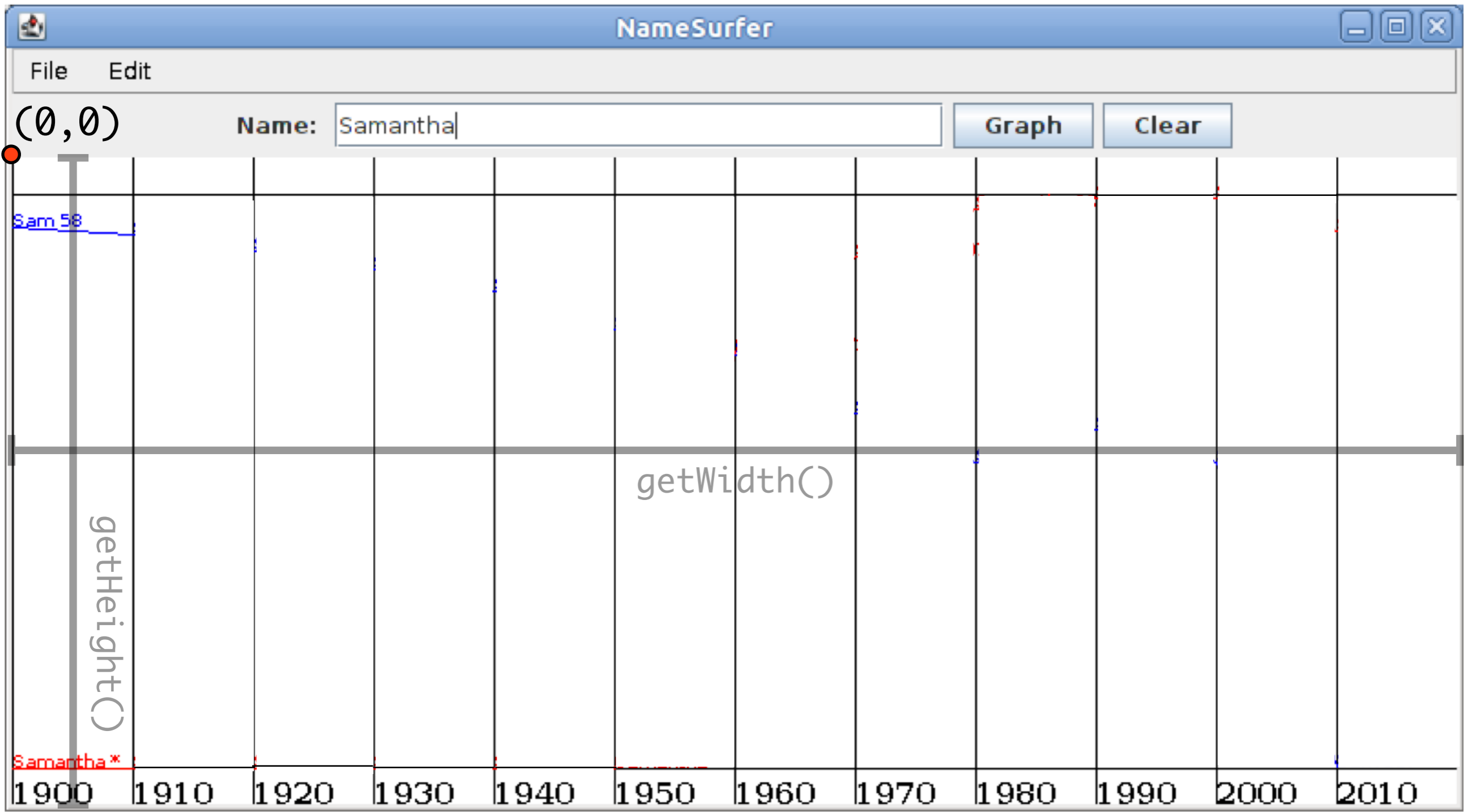
/** The first decade in the database */
public static final int START_DECADE = 1900;

/** The number of decades */
public static final int NDECADES = 12;

/** The maximum rank in the database */
public static final int MAX_RANK = 1000;

/** The number of pixels to reserve at the top and bottom */
public static final int GRAPH_MARGIN_SIZE = 20;

```

entries

Sam NameSurferEntry

(58 69 99 131 168 236 278 380 467 408 466 997)

Samantha NameSurferEntry

(0 0 0 0 0 272 107 26 5 7 63)

from NameSurferConstants

```

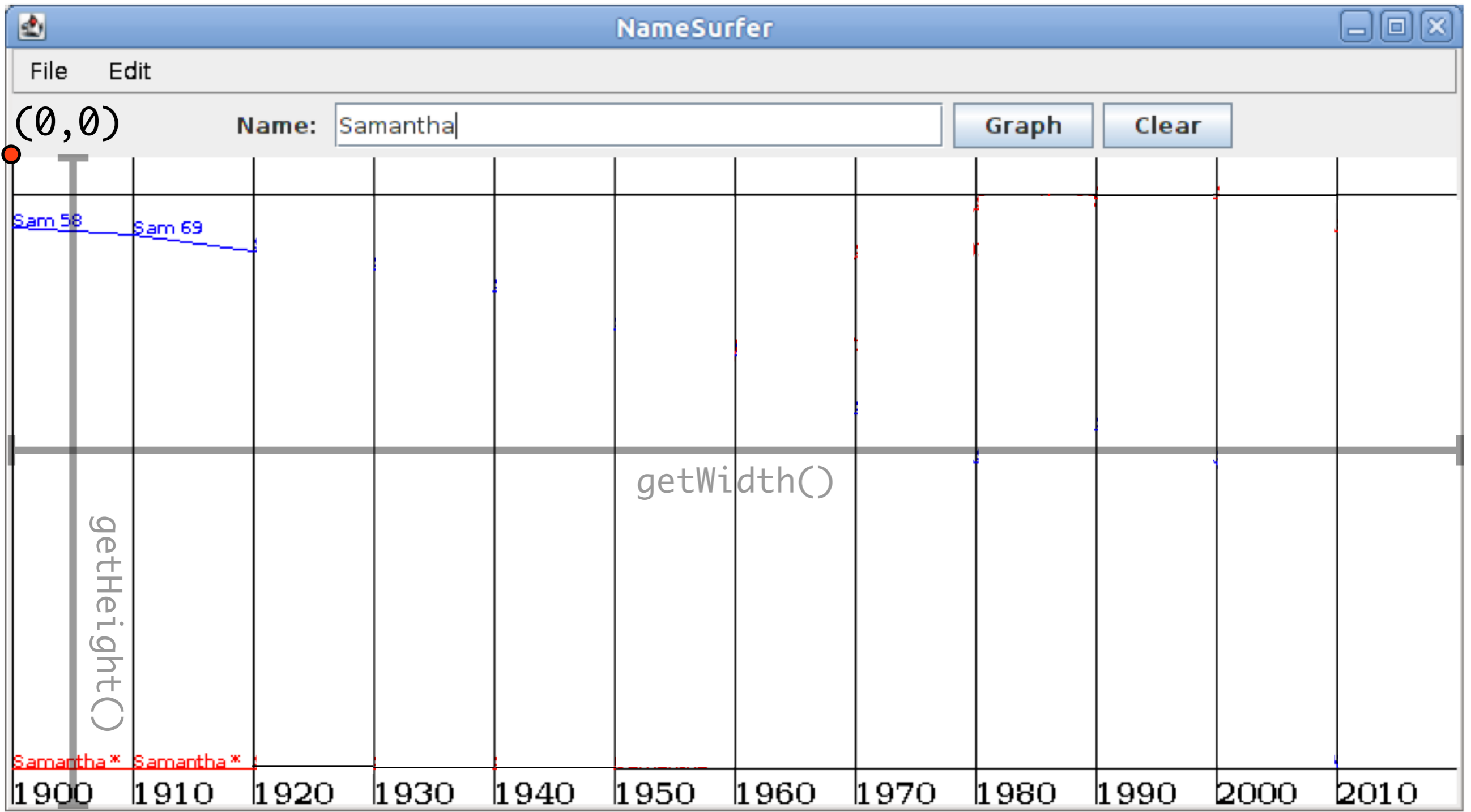
/** The first decade in the database */
public static final int START_DECADE = 1900;

/** The number of decades */
public static final int NDECADES = 12;

/** The maximum rank in the database */
public static final int MAX_RANK = 1000;

/** The number of pixels to reserve at the top and bottom */
public static final int GRAPH_MARGIN_SIZE = 20;

```



entries

Sam NameSurferEntry (58 69 99 131 168 236 278 380 467 408 466 997)
Samantha NameSurferEntry (0 0 0 0 0 272 107 26 5 7 63)

from NameSurferConstants

```

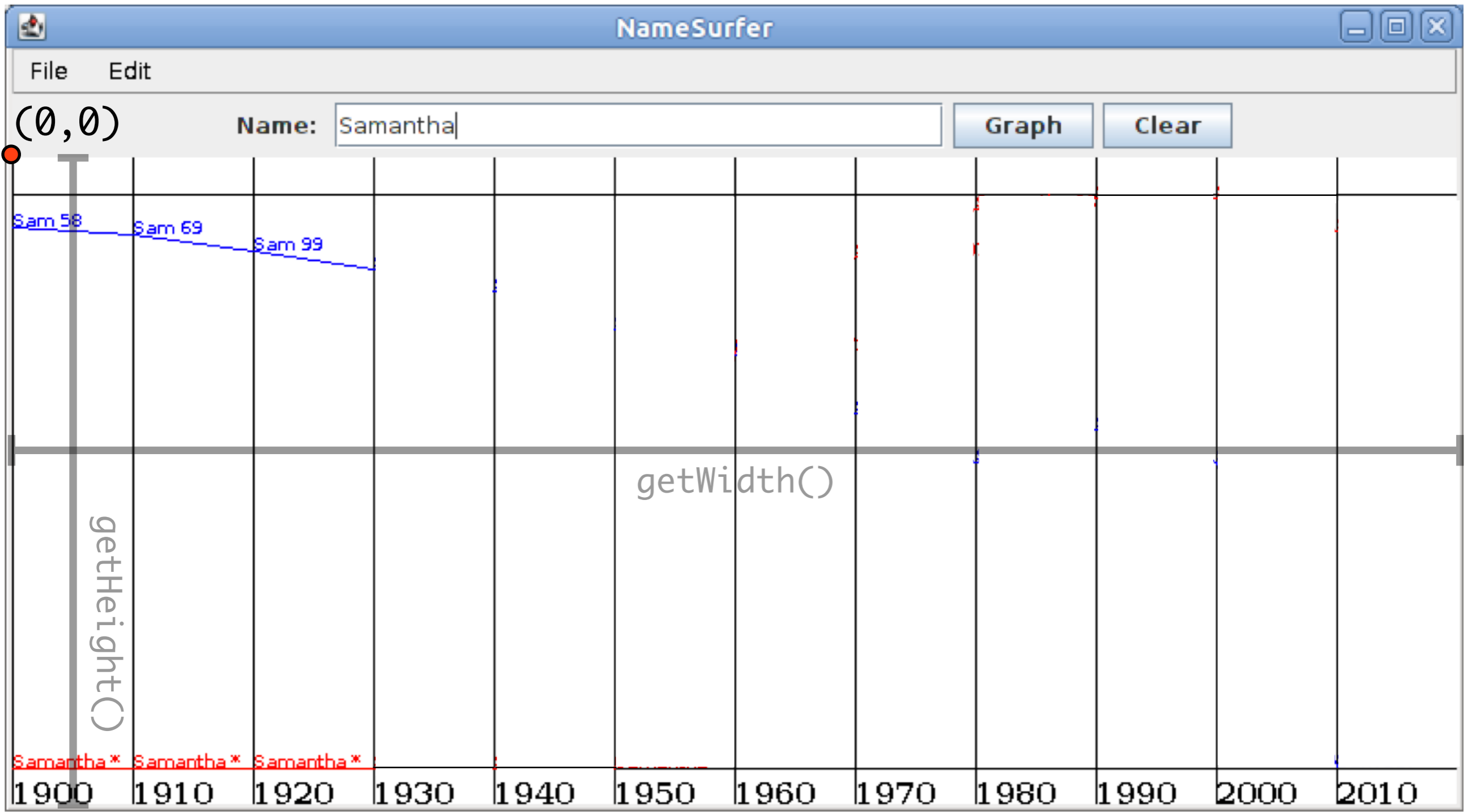
/** The first decade in the database */
public static final int START_DECADE = 1900;

/** The number of decades */
public static final int NDECADES = 12;

/** The maximum rank in the database */
public static final int MAX_RANK = 1000;

/** The number of pixels to reserve at the top and bottom */
public static final int GRAPH_MARGIN_SIZE = 20;

```



entries

Sam NameSurferEntry

(58 69 99 131 168 236 278 380 467 408 466 997)

Samantha NameSurferEntry

(0 0 0 0 0 272 107 26 5 7 63)

from NameSurferConstants

```

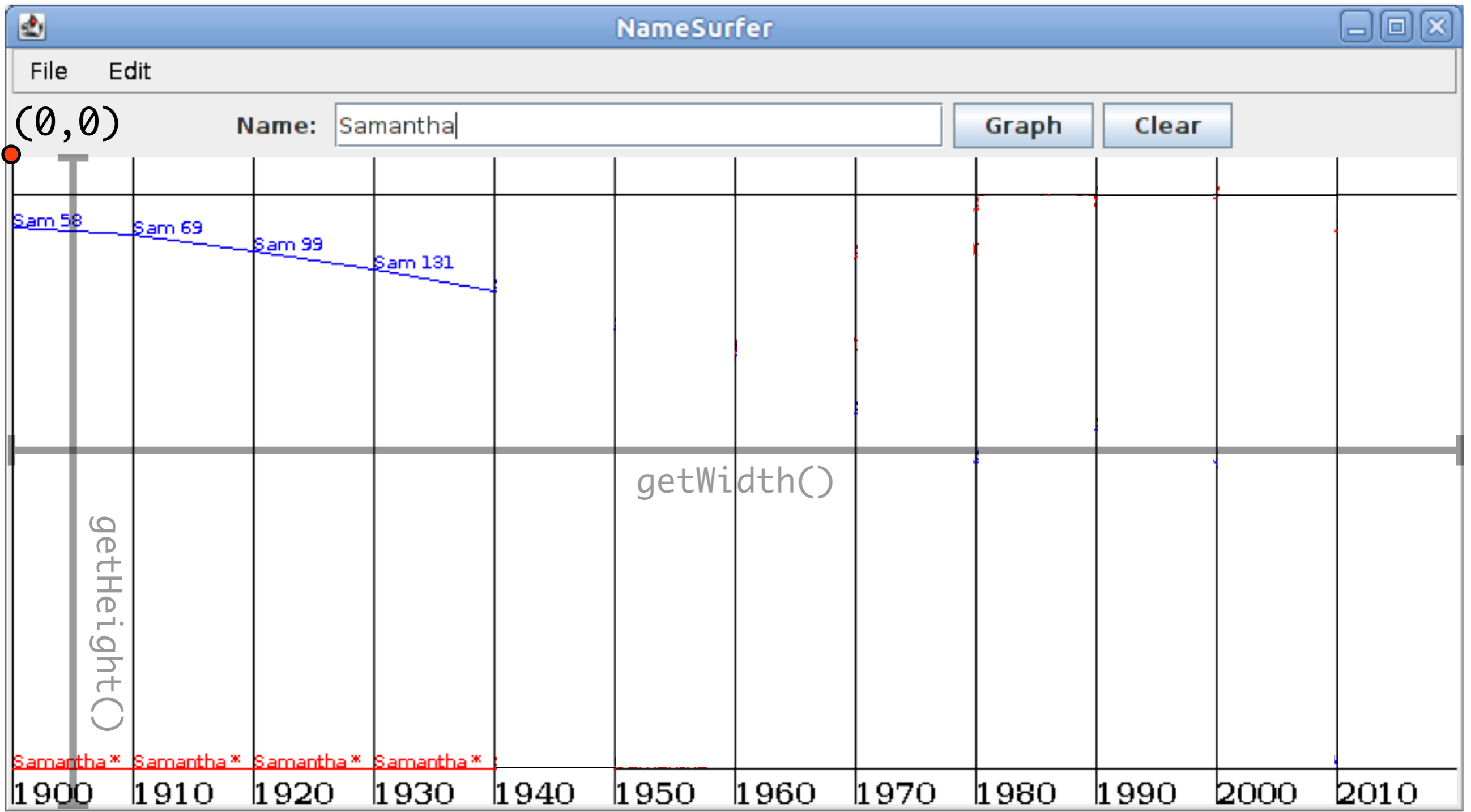
/** The first decade in the database */
public static final int START_DECADE = 1900;

/** The number of decades */
public static final int NDECADES = 12;

/** The maximum rank in the database */
public static final int MAX_RANK = 1000;

/** The number of pixels to reserve at the top and bottom */
public static final int GRAPH_MARGIN_SIZE = 20;

```



entries

Sam NameSurferEntry

(58 69 99 131 168 236 278 380 467 408 466 997)

Samantha NameSurferEntry

(0 0 0 0 0 272 107 26 5 7 63)

from NameSurferConstants

```

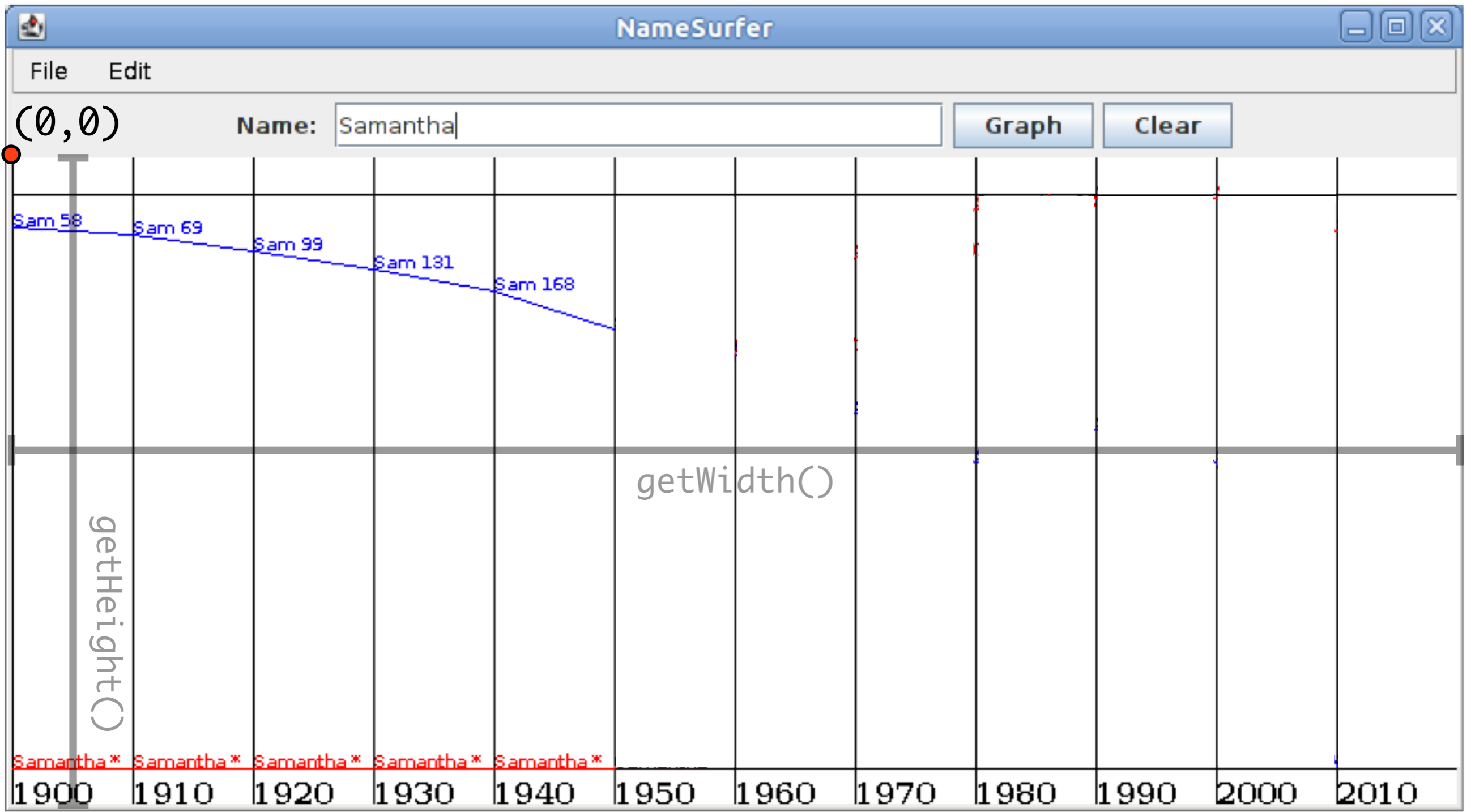
/** The first decade in the database */
public static final int START_DECADE = 1900;

/** The number of decades */
public static final int NDECADES = 12;

/** The maximum rank in the database */
public static final int MAX_RANK = 1000;

/** The number of pixels to reserve at the top and bottom */
public static final int GRAPH_MARGIN_SIZE = 20;

```



entries

Sam NameSurferEntry (58 69 99 131 168 236 278 380 467 408 466 997)
Samantha NameSurferEntry (0 0 0 0 0 272 107 26 5 7 63)

from NameSurferConstants

```

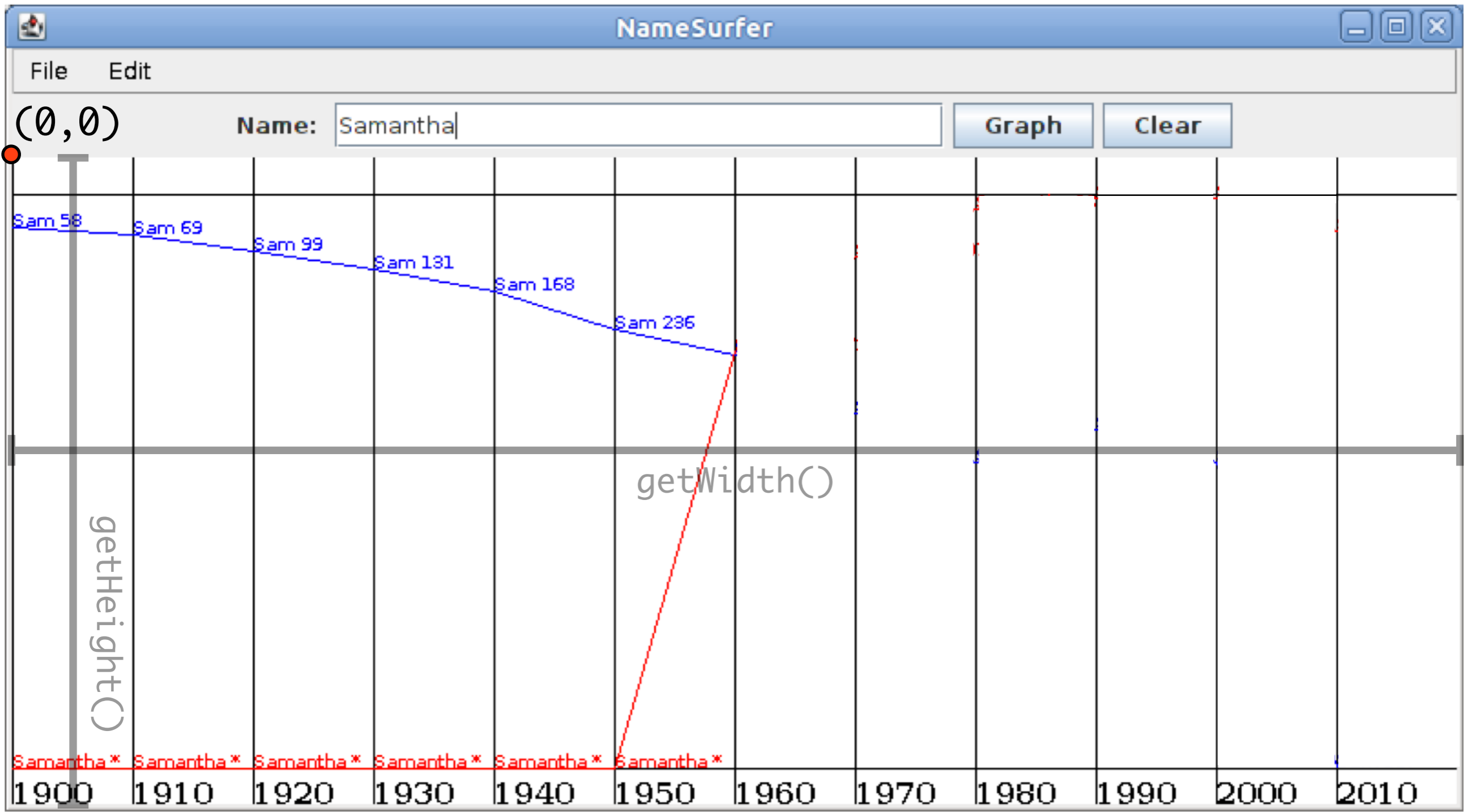
/** The first decade in the database */
public static final int START_DECADE = 1900;

/** The number of decades */
public static final int NDECADES = 12;

/** The maximum rank in the database */
public static final int MAX_RANK = 1000;

/** The number of pixels to reserve at the top and bottom */
public static final int GRAPH_MARGIN_SIZE = 20;

```



entries

Sam NameSurferEntry (58 69 99 131 168 236 278 380 467 408 466 997)
Samantha NameSurferEntry (0 0 0 0 0 272 107 26 5 7 63)

from NameSurferConstants

```

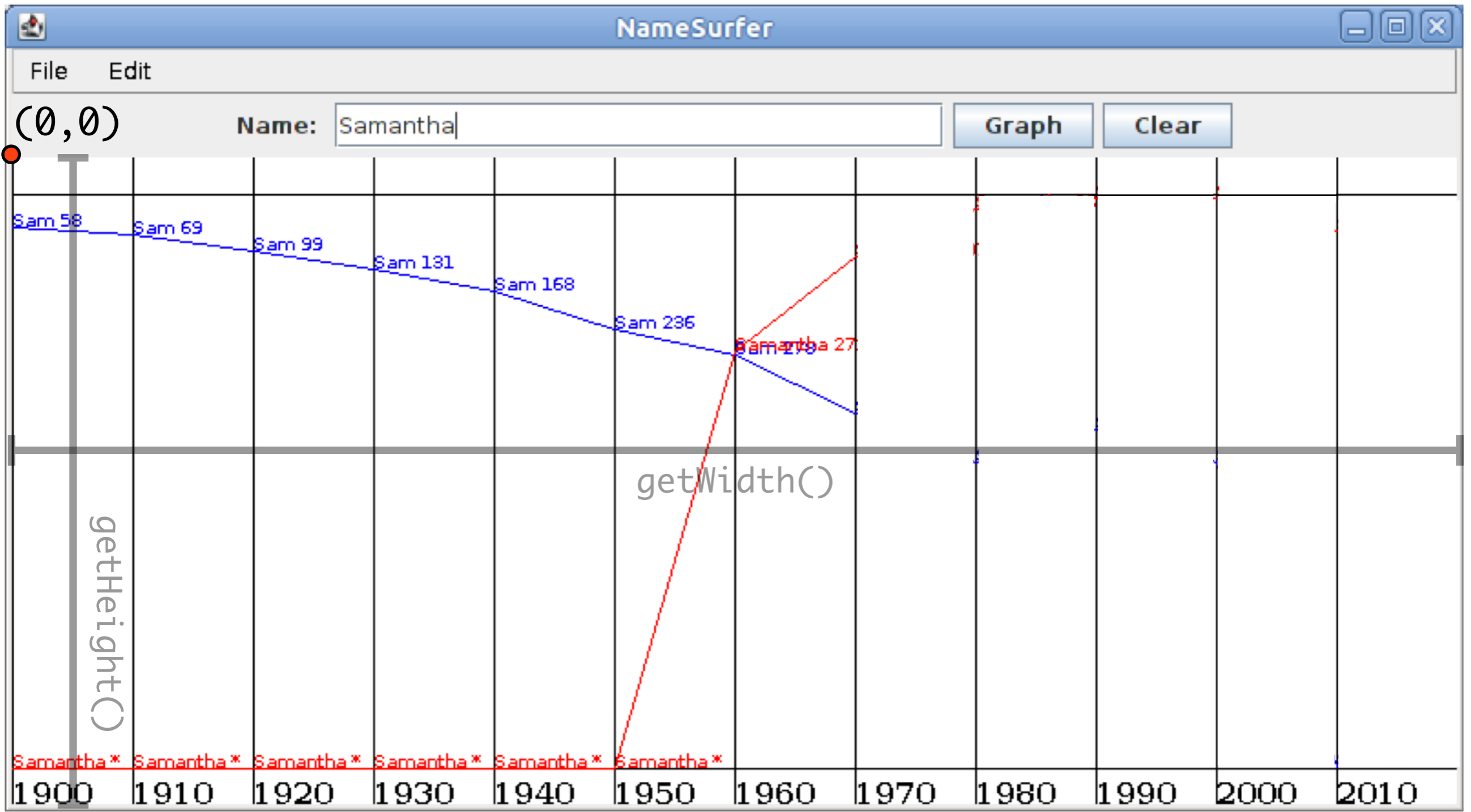
/** The first decade in the database */
public static final int START_DECADE = 1900;

/** The number of decades */
public static final int NDECADES = 12;

/** The maximum rank in the database */
public static final int MAX_RANK = 1000;

/** The number of pixels to reserve at the top and bottom */
public static final int GRAPH_MARGIN_SIZE = 20;

```



entries

Sam NameSurferEntry

(58 69 99 131 168 236 278 380 467 408 466 997)

Samantha NameSurferEntry

(0 0 0 0 0 272 107 26 5 7 63)

from NameSurferConstants

```

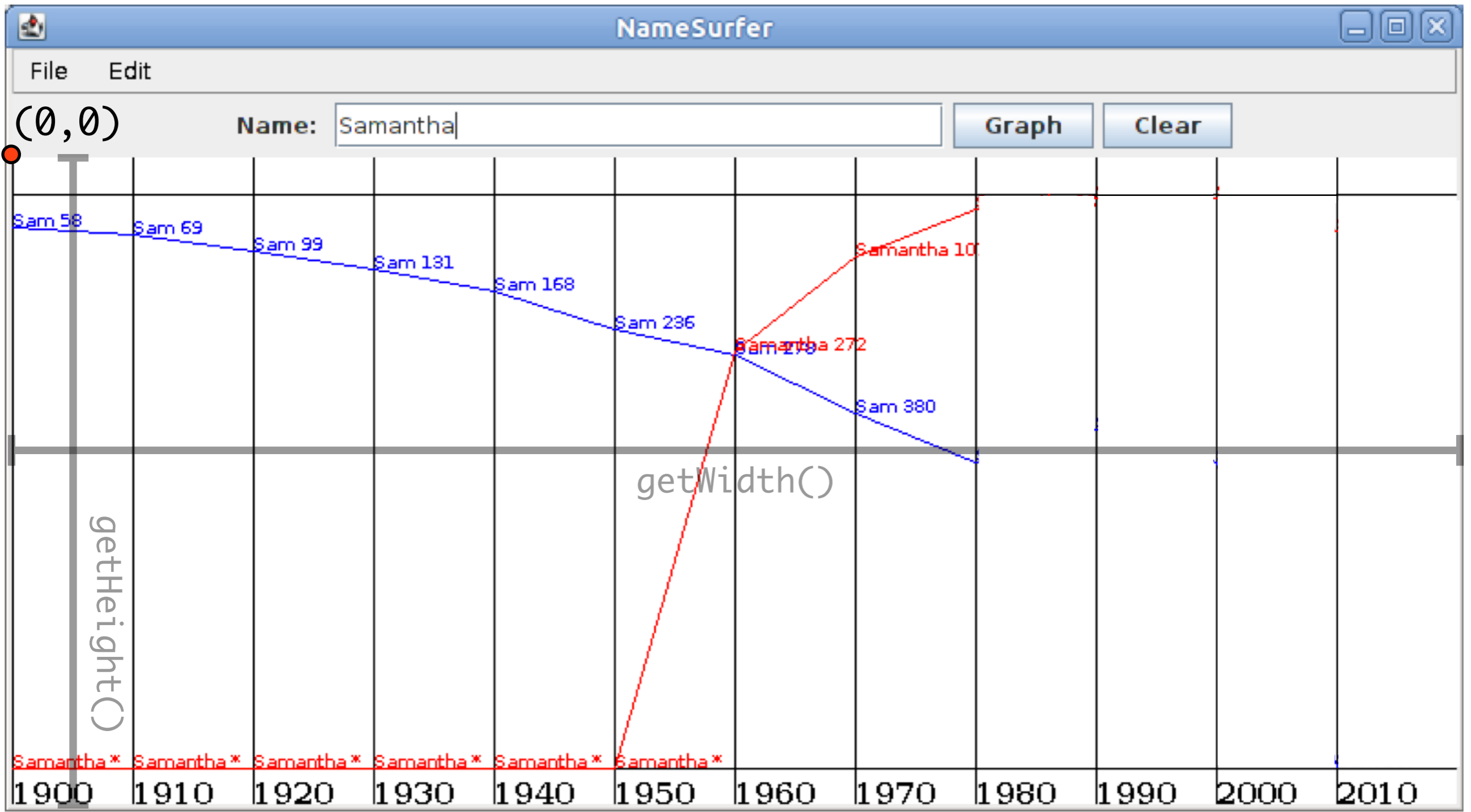
/** The first decade in the database */
public static final int START_DECADE = 1900;

/** The number of decades */
public static final int NDECADES = 12;

/** The maximum rank in the database */
public static final int MAX_RANK = 1000;

/** The number of pixels to reserve at the top and bottom */
public static final int GRAPH_MARGIN_SIZE = 20;

```



entries

Sam NameSurferEntry (58 69 99 131 168 236 278 380 467 408 466 997)
Samantha NameSurferEntry (0 0 0 0 0 272 107 26 5 7 63)

from NameSurferConstants

```

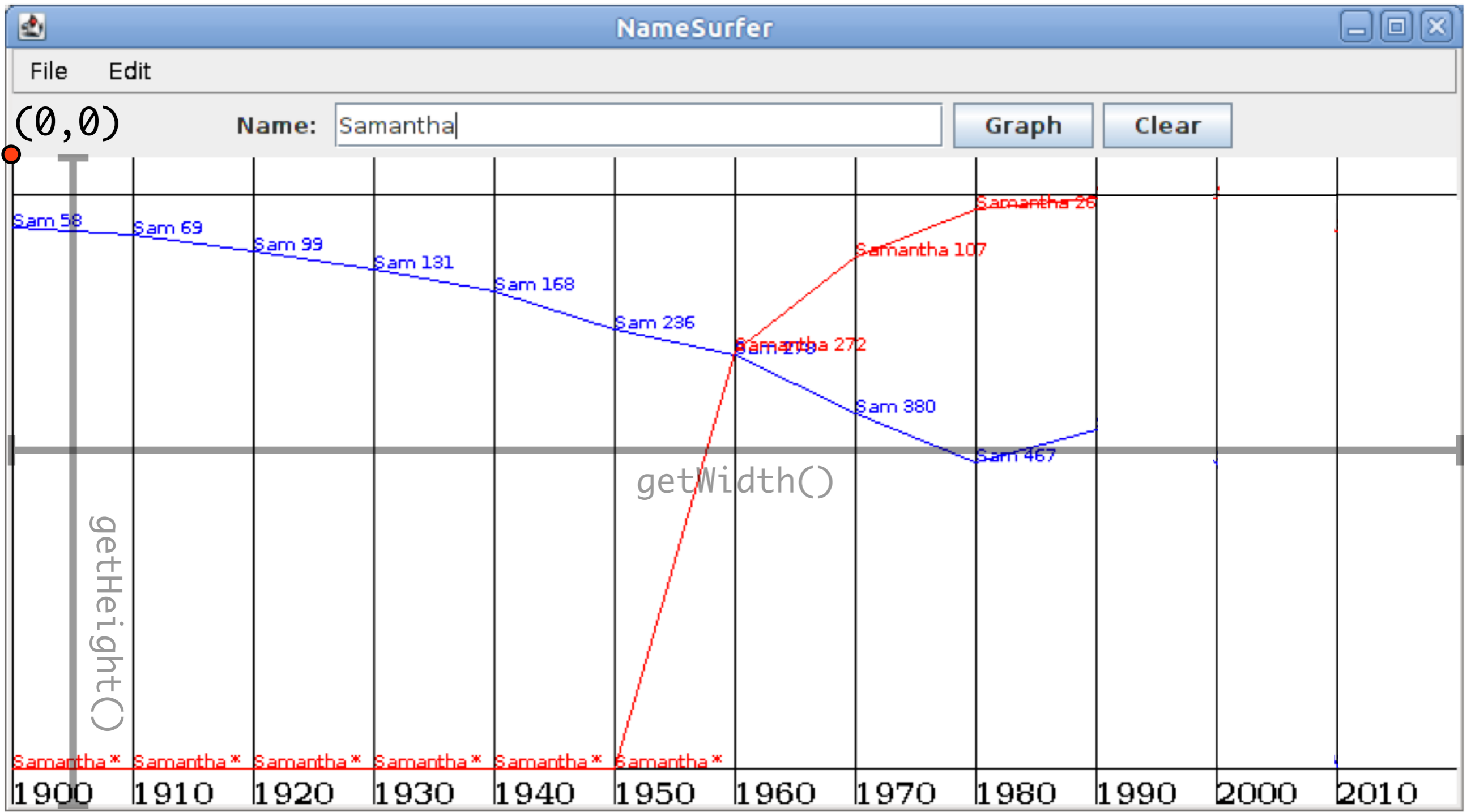
/** The first decade in the database */
public static final int START_DECADE = 1900;

/** The number of decades */
public static final int NDECADES = 12;

/** The maximum rank in the database */
public static final int MAX_RANK = 1000;

/** The number of pixels to reserve at the top and bottom */
public static final int GRAPH_MARGIN_SIZE = 20;

```

entries

Sam NameSurferEntry

(58 69 99 131 168 236 278 380 467 408 466 997)

Samantha NameSurferEntry

(0 0 0 0 0 0 272 107 26 5 7 63)

from NameSurferConstants

```

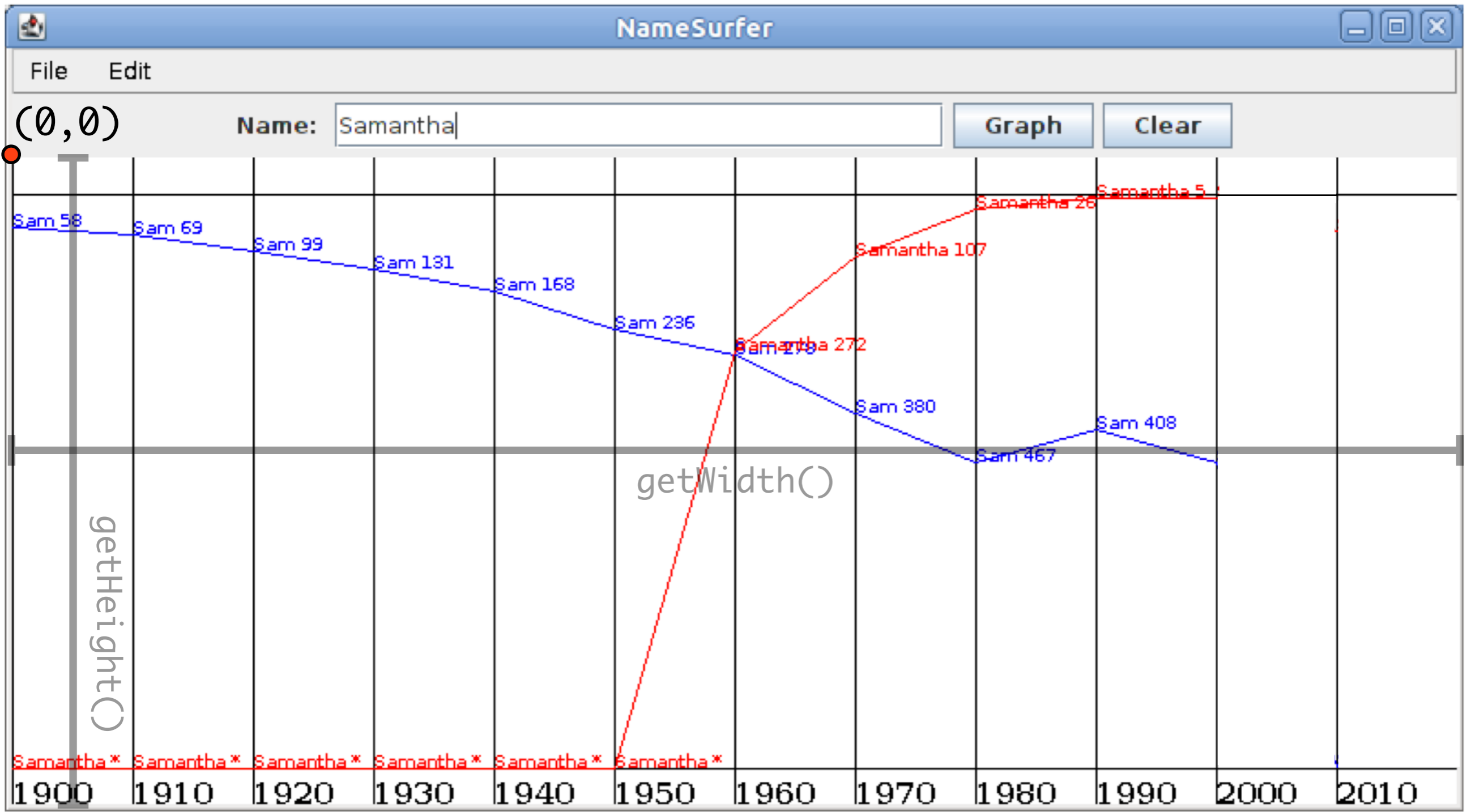
/** The first decade in the database */
public static final int START_DECADE = 1900;

/** The number of decades */
public static final int NDECADES = 12;

/** The maximum rank in the database */
public static final int MAX_RANK = 1000;

/** The number of pixels to reserve at the top and bottom */
public static final int GRAPH_MARGIN_SIZE = 20;

```



entries

Sam NameSurferEntry (58 69 99 131 168 236 278 380 467 408 466 997)
Samantha NameSurferEntry (0 0 0 0 0 0 272 107 26 5 7 63)

from NameSurferConstants

```

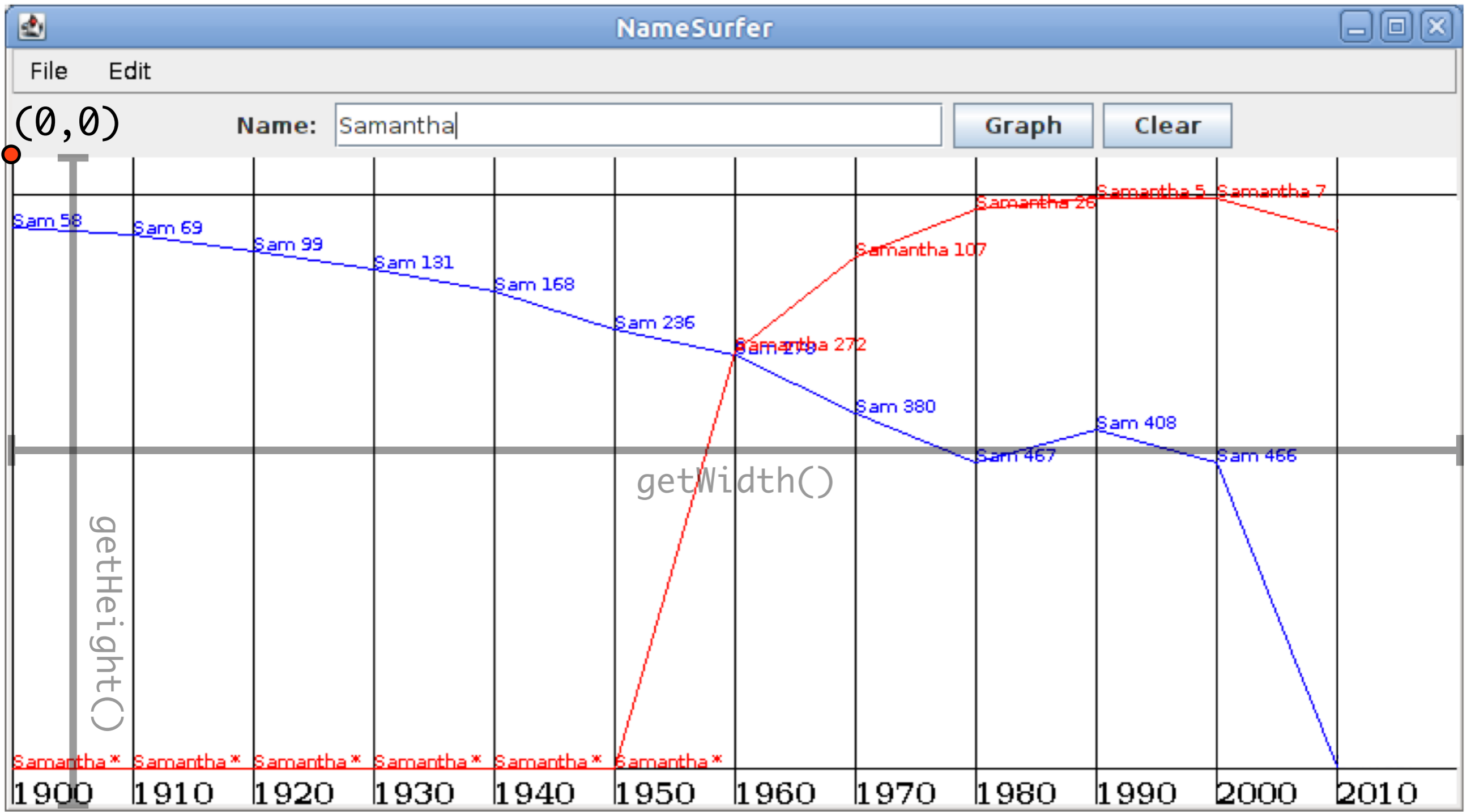
/** The first decade in the database */
public static final int START_DECADE = 1900;

/** The number of decades */
public static final int NDECADES = 12;

/** The maximum rank in the database */
public static final int MAX_RANK = 1000;

/** The number of pixels to reserve at the top and bottom */
public static final int GRAPH_MARGIN_SIZE = 20;

```



entries

Sam NameSurferEntry

(58 69 99 131 168 236 278 380 467 408 466 997)

Samantha NameSurferEntry

(0 0 0 0 0 272 107 26 5 7 63)

from NameSurferConstants

```

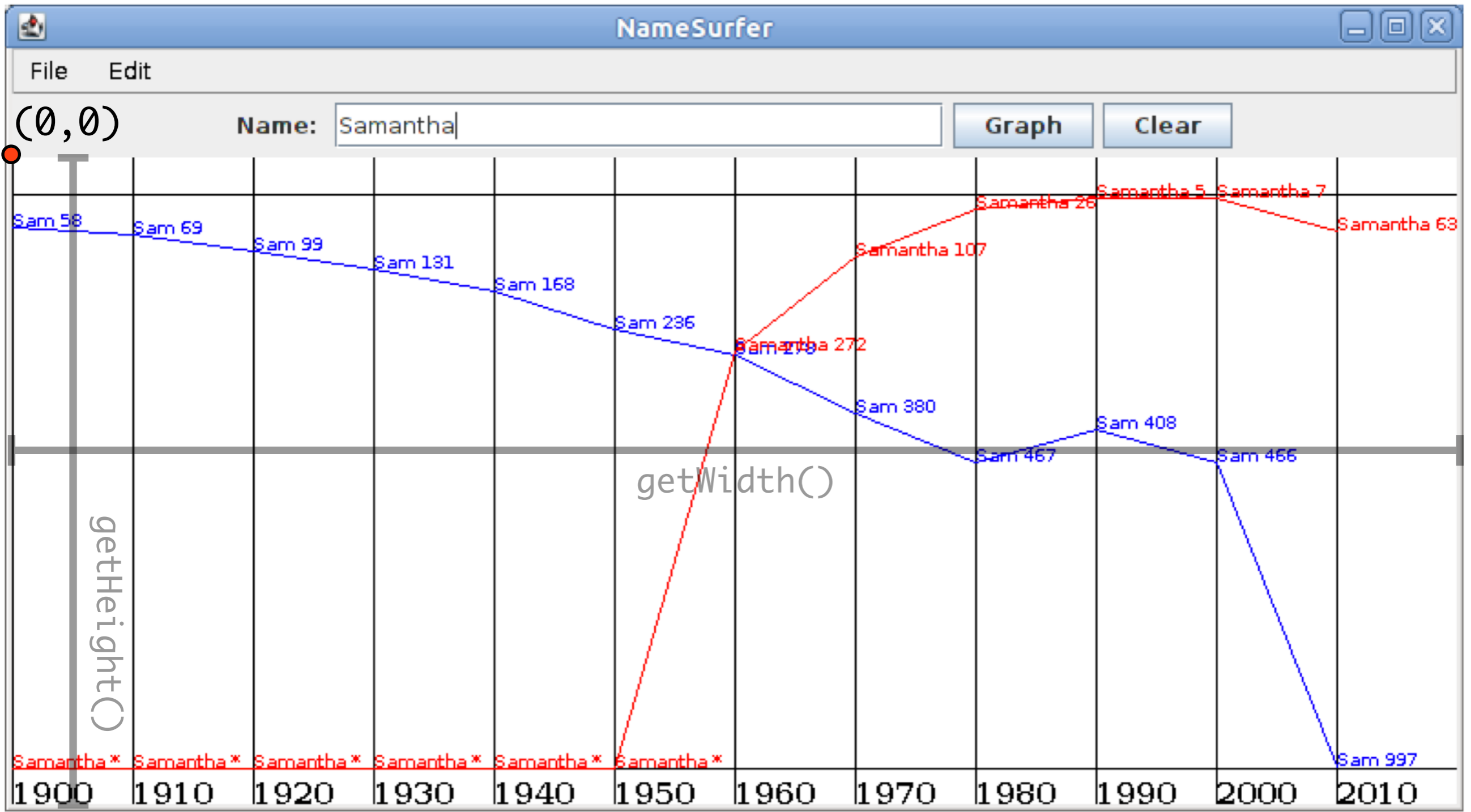
/** The first decade in the database */
public static final int START_DECADE = 1900;

/** The number of decades */
public static final int NDECADES = 12;

/** The maximum rank in the database */
public static final int MAX_RANK = 1000;

/** The number of pixels to reserve at the top and bottom */
public static final int GRAPH_MARGIN_SIZE = 20;

```



entries

Sam NameSurferEntry
(58 69 99 131 168 236 278 380 467 408 466 997)
Samantha NameSurferEntry
(0 0 0 0 0 0 272 107 26 5 7 63)

from NameSurferConstants

```

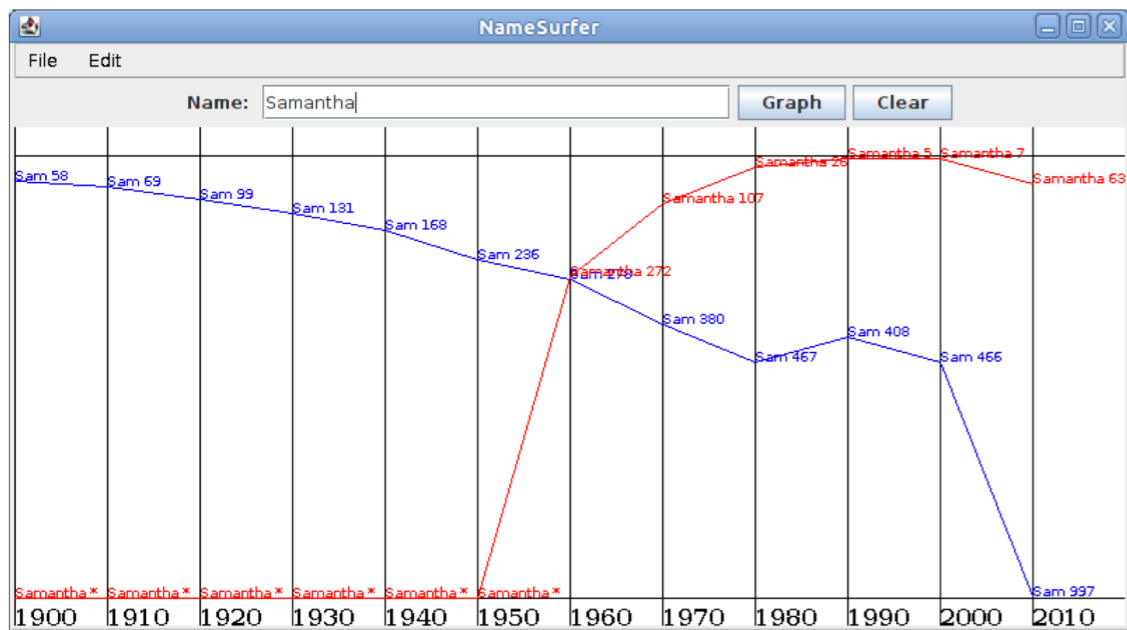
/** The first decade in the database */
public static final int START_DECADE = 1900;

/** The number of decades */
public static final int NDECADES = 12;

/** The maximum rank in the database */
public static final int MAX_RANK = 1000;

/** The number of pixels to reserve at the top and bottom */
public static final int GRAPH_MARGIN_SIZE = 20;

```



entries

<p>Sam NameSurferEntry (58 69 99 131 168 236 278 380 467 408 466 997)</p>
<p>Samantha NameSurferEntry (0 0 0 0 0 0 272 107 26 5 7 63)</p>

from NameSurferConstants

```

/** The first decade in the database */
public static final int START_DECADE = 1900;

/** The number of decades */
public static final int NDECADES = 12;

/** The maximum rank in the database */
public static final int MAX_RANK = 1000;

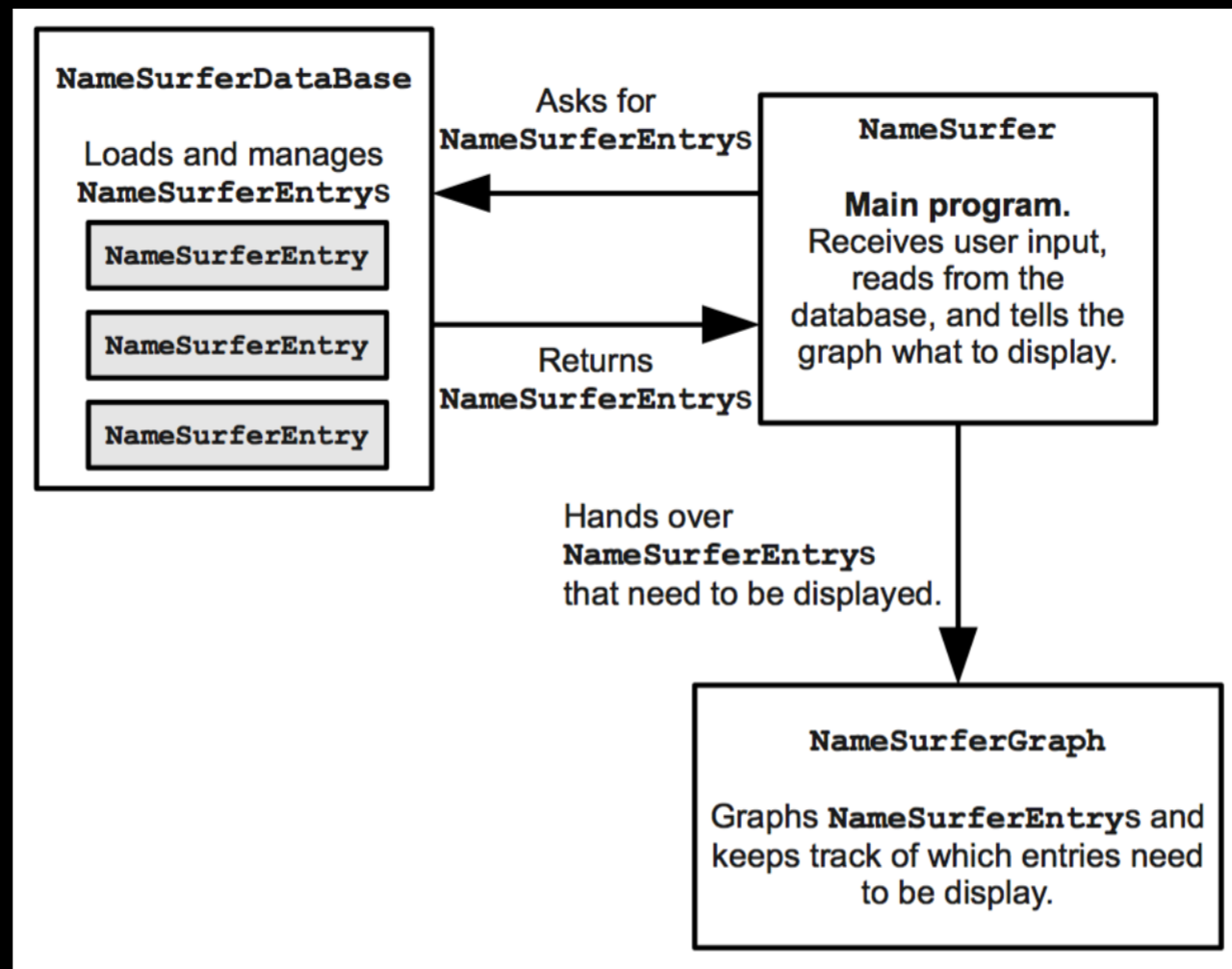
/** The number of pixels to reserve at the top and bottom */
public static final int GRAPH_MARGIN_SIZE = 20;

```

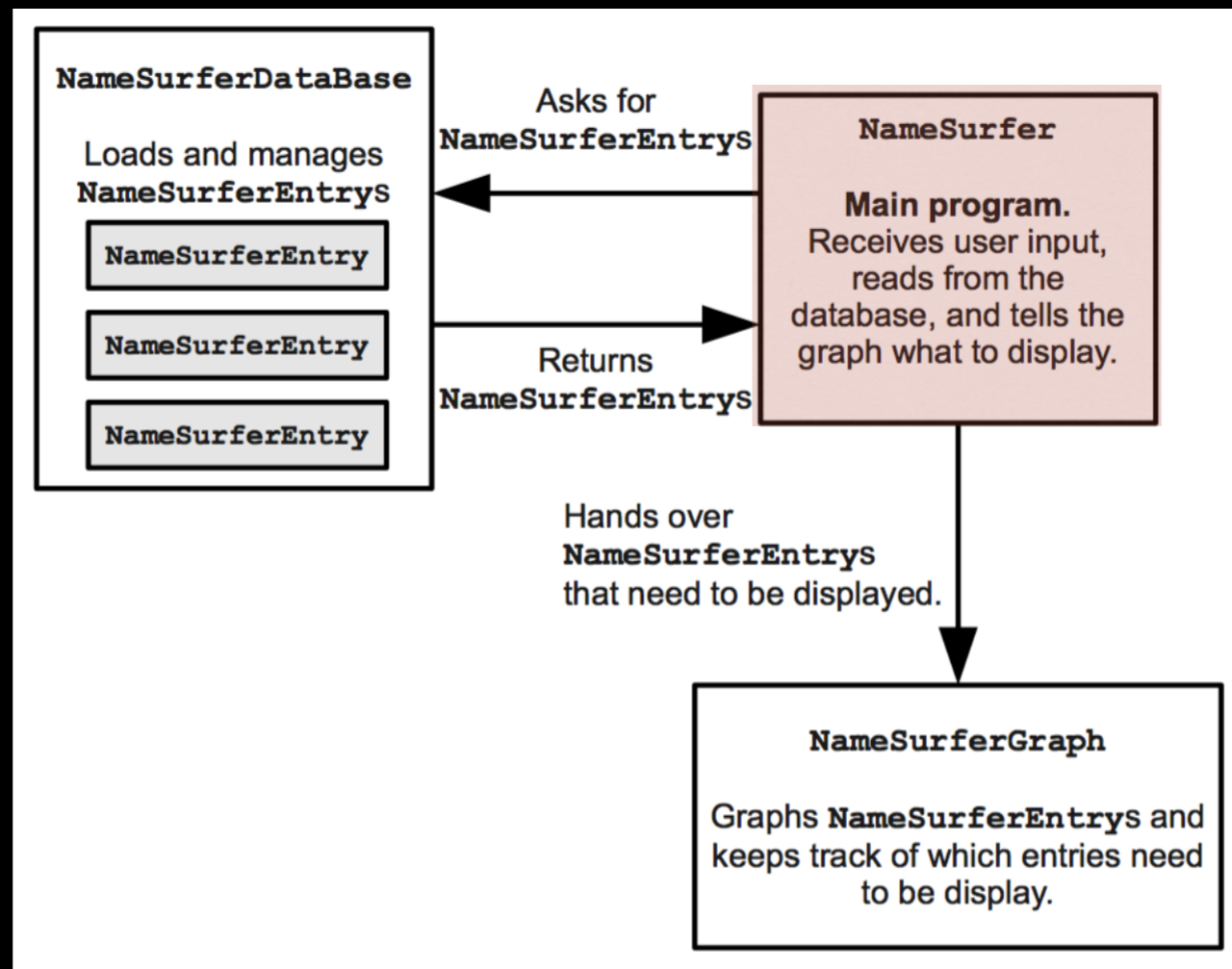
NameSurferConstants

```
public interface NameSurferConstants {  
  
    /** The width of the application window */  
    public static final int APPLICATION_WIDTH = 800;  
  
    /** The height of the application window */  
    public static final int APPLICATION_HEIGHT = 600;  
  
    /** The name of the file containing the data */  
    public static final String NAMES_DATA_FILE = "names-data.txt";  
  
    /** The first decade in the database */  
    public static final int START_DECADE = 1900;  
  
    /** The number of decades */  
    public static final int NDECADES = 12;  
  
    /** The maximum rank in the database */  
    public static final int MAX_RANK = 1000;  
  
    /** The number of pixels to reserve at the top and bottom */  
    public static final int GRAPH_MARGIN_SIZE = 20;  
  
}
```

NameSurfer



NameSurfer



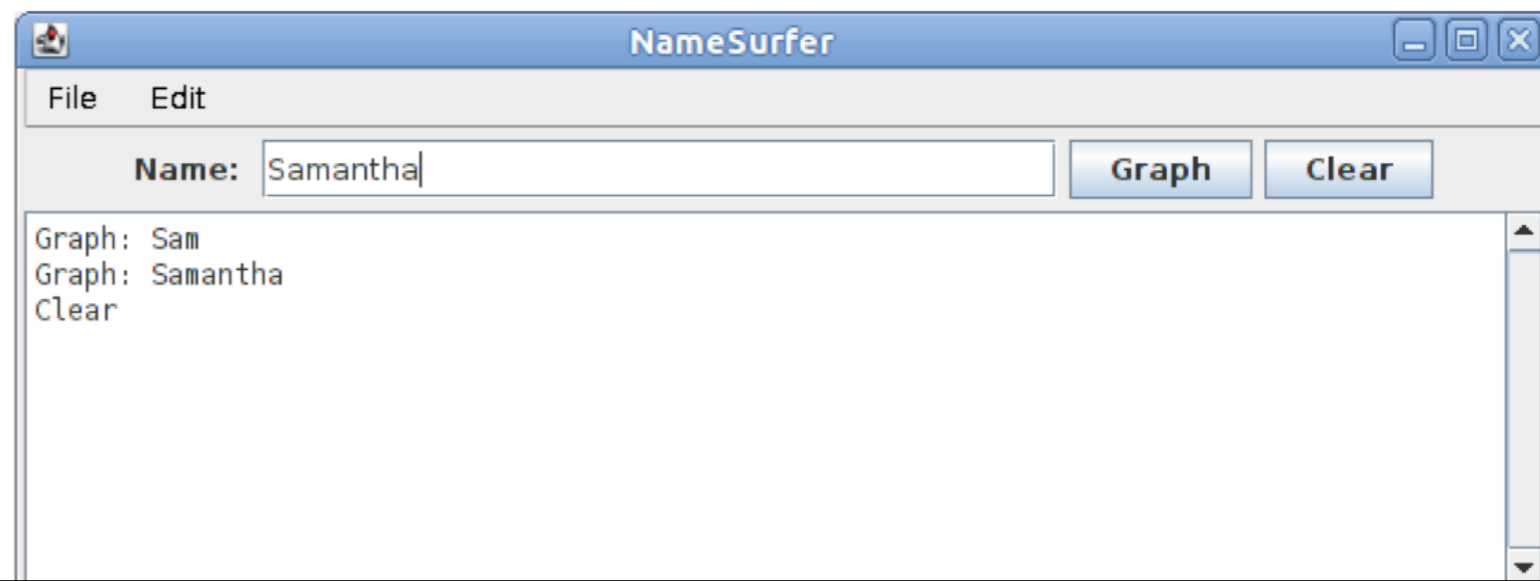
```

public class NameSurfer extends Program implements NameSurferConstants {

    /* Method: init() */
    /**
     * This method has the responsibility for reading in the data base
     * and initializing the interactors at the top of the window.
     */
    public void init() {
        // You fill this in, along with any helper methods //
    }

    /* Method: actionPerformed(e) */
    /**
     * This class is responsible for detecting when the buttons are
     * clicked, so you will have to define a method to respond to
     * button actions.
     */
    public void actionPerformed(ActionEvent e) {
        // You fill this in //
    }
}

```



To add the graph to the screen

```
private NameSurferGraph graph;
```

```
graph = new NameSurferGraph();  
add(graph);
```

To add the graph to the screen

```
private NameSurferGraph graph;
```

```
graph = new NameSurferGraph();  
add(graph);
```

init()

NameSurferDataBase

Loads and manages
NameSurferEntryS

NameSurferDataBase

Loads and manages
NameSurferEntryS

NameSurferEntry

NameSurferEntry

NameSurferEntry

NamesData.txt

```
...  
Sam 58 69 99 131 168 236 278 380 467 408 466  
Samantha 0 0 0 0 0 0 272 107 26 5 7  
Samara 0 0 0 0 0 0 0 0 0 0 886  
Samir 0 0 0 0 0 0 0 0 920 0 798  
Sammie 537 545 351 325 333 396 565 772 930 0 0  
Sammy 0 887 544 299 202 262 321 395 575 639 755  
Samson 0 0 0 0 0 0 0 0 0 0 915  
Samuel 31 41 46 60 61 71 83 61 52 35 28  
Sandi 0 0 0 0 704 864 621 695 0 0 0  
Sandra 0 942 606 50 6 12 11 39 94 168 257  
...
```

NameSurferDataBase

Loads and manages
NameSurferEntryS

NameSurferEntry

NameSurferEntry

NameSurferEntry

NamesData.txt

```
...  
Sam 58 69 99 131 168 236 278 380 467 408 466  
Samantha 0 0 0 0 0 0 272 107 26 5 7  
Samara 0 0 0 0 0 0 0 0 0 0 886  
Samir 0 0 0 0 0 0 0 0 920 0 798  
Sammie 537 545 351 325 333 396 565 772 930 0 0  
Sammy 0 887 544 299 202 262 321 395 575 639 755  
Samson 0 0 0 0 0 0 0 0 0 0 915  
Samuel 31 41 46 60 61 71 83 61 52 35 28  
Sandi 0 0 0 0 704 864 621 695 0 0 0  
Sandra 0 942 606 50 6 12 11 39 94 168 257  
...
```

NameSurferDataBase

Loads and manages
NameSurferEntryS

NameSurferEntry

NameSurferEntry

NameSurferEntry

NamesData.txt

```
...  
Sam 58 69 99 131 168 236 278 380 467 408 466  
Samantha 0 0 0 0 0 0 272 107 26 5 7  
Samara 0 0 0 0 0 0 0 0 0 0 886  
Samir 0 0 0 0 0 0 0 0 920 0 798  
Sammie 537 545 351 325 333 396 565 772 930 0 0  
Sammy 0 887 544 299 202 262 321 395 575 639 755  
Samson 0 0 0 0 0 0 0 0 0 0 915  
Samuel 31 41 46 60 61 71 83 61 52 35 28  
Sandi 0 0 0 0 704 864 621 695 0 0 0  
Sandra 0 942 606 50 6 12 11 39 94 168 257  
...
```

NameSurferDataBase

Loads and manages
NameSurferEntryS

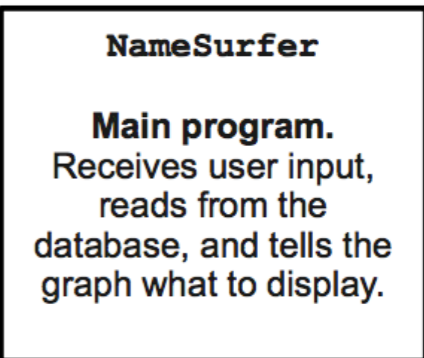
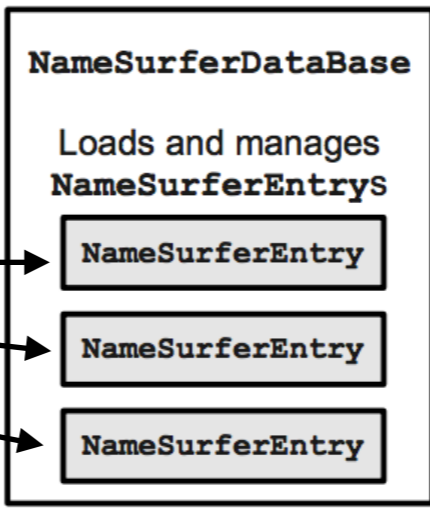
NameSurferEntry

NameSurferEntry

NameSurferEntry

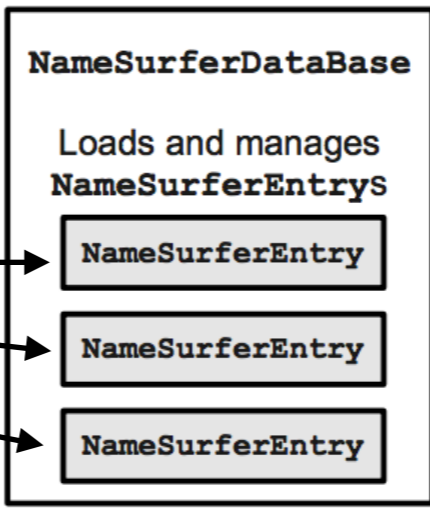
NamesData.txt

```
...  
Sam 58 69 99 131 168 236 278 380 467 408 466  
Samantha 0 0 0 0 0 0 272 107 26 5 7  
Samara 0 0 0 0 0 0 0 0 0 0 886  
Samir 0 0 0 0 0 0 0 0 920 0 798  
Sammie 537 545 351 325 333 396 565 772 930 0 0  
Sammy 0 887 544 299 202 262 321 395 575 639 755  
Samson 0 0 0 0 0 0 0 0 0 0 915  
Samuel 31 41 46 60 61 71 83 61 52 35 28  
Sandi 0 0 0 0 704 864 621 695 0 0 0  
Sandra 0 942 606 50 6 12 11 39 94 168 257  
...
```

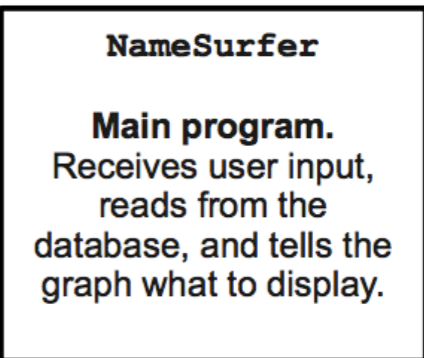


NamesData.txt

```
...  
Sam 58 69 99 131 168 236 278 380 467 408 466  
Samantha 0 0 0 0 0 0 272 107 26 5 7  
Samara 0 0 0 0 0 0 0 0 0 0 886  
Samir 0 0 0 0 0 0 0 0 920 0 798  
Sammie 537 545 351 325 333 396 565 772 930 0 0  
Sammy 0 887 544 299 202 262 321 395 575 639 755  
Samson 0 0 0 0 0 0 0 0 0 0 915  
Samuel 31 41 46 60 61 71 83 61 52 35 28  
Sandi 0 0 0 0 704 864 621 695 0 0 0  
Sandra 0 942 606 50 6 12 11 39 94 168 257  
...
```

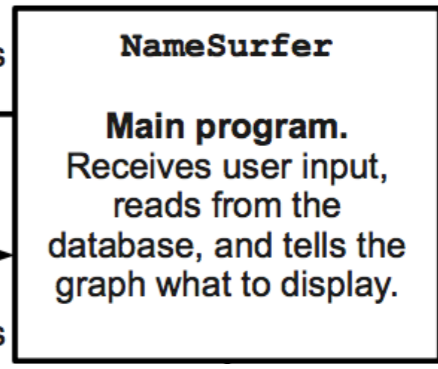
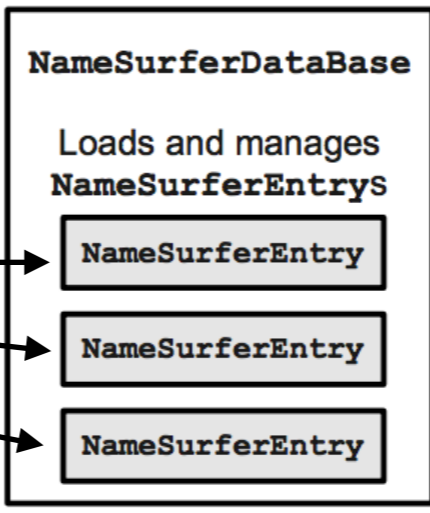


Asks for
NameSurferEntrys



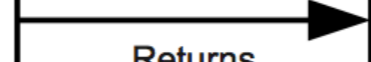
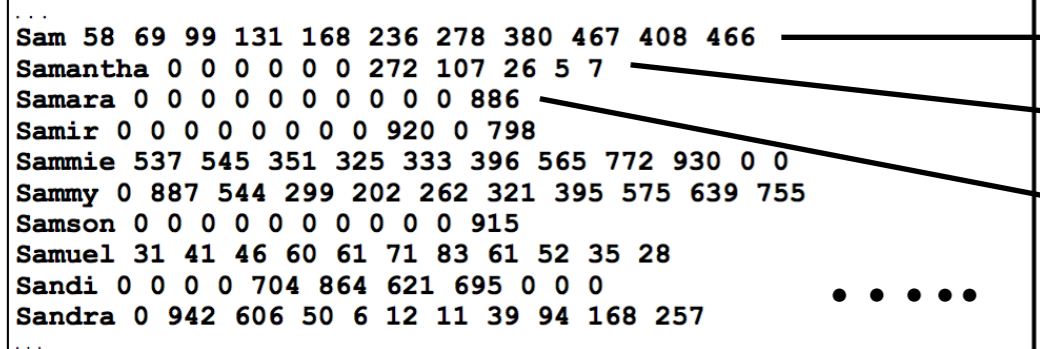
NamesData.txt

```
...  
Sam 58 69 99 131 168 236 278 380 467 408 466  
Samantha 0 0 0 0 0 0 272 107 26 5 7  
Samara 0 0 0 0 0 0 0 0 0 0 886  
Samir 0 0 0 0 0 0 0 0 920 0 798  
Sammie 537 545 351 325 333 396 565 772 930 0 0  
Sammy 0 887 544 299 202 262 321 395 575 639 755  
Samson 0 0 0 0 0 0 0 0 0 0 915  
Samuel 31 41 46 60 61 71 83 61 52 35 28  
Sandi 0 0 0 0 704 864 621 695 0 0 0  
Sandra 0 942 606 50 6 12 11 39 94 168 257  
...
```



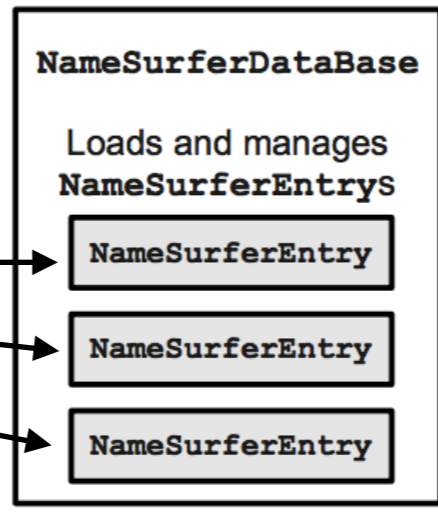
Asks for
NameSurferEntrys

Returns
NameSurferEntrys

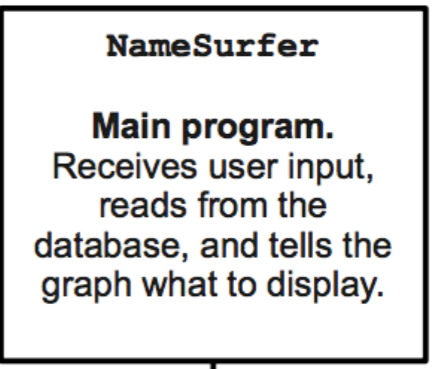


NamesData.txt

```
...  
Sam 58 69 99 131 168 236 278 380 467 408 466  
Samantha 0 0 0 0 0 0 272 107 26 5 7  
Samara 0 0 0 0 0 0 0 0 0 0 886  
Samir 0 0 0 0 0 0 0 0 920 0 798  
Sammie 537 545 351 325 333 396 565 772 930 0 0  
Sammy 0 887 544 299 202 262 321 395 575 639 755  
Samson 0 0 0 0 0 0 0 0 0 0 915  
Samuel 31 41 46 60 61 71 83 61 52 35 28  
Sandi 0 0 0 0 704 864 621 695 0 0 0  
Sandra 0 942 606 50 6 12 11 39 94 168 257  
...
```

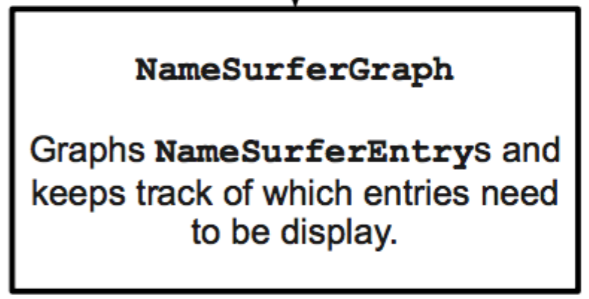


Asks for
NameSurferEntrys



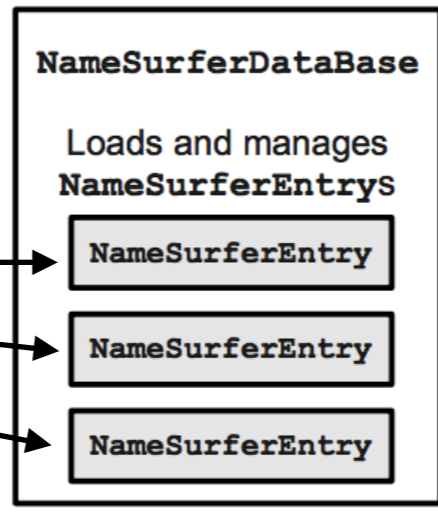
Returns
NameSurferEntrys

Hands over
NameSurferEntrys
that need to be displayed.

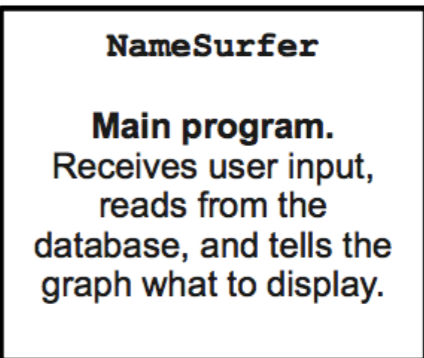


NamesData.txt

```
...
Sam 58 69 99 131 168 236 278 380 467 408 466
Samantha 0 0 0 0 0 0 272 107 26 5 7
Samara 0 0 0 0 0 0 0 0 0 0 886
Samir 0 0 0 0 0 0 0 0 920 0 798
Sammie 537 545 351 325 333 396 565 772 930 0 0
Sammy 0 887 544 299 202 262 321 395 575 639 755
Samson 0 0 0 0 0 0 0 0 0 0 915
Samuel 31 41 46 60 61 71 83 61 52 35 28
Sandi 0 0 0 0 704 864 621 695 0 0 0
Sandra 0 942 606 50 6 12 11 39 94 168 257
...
```

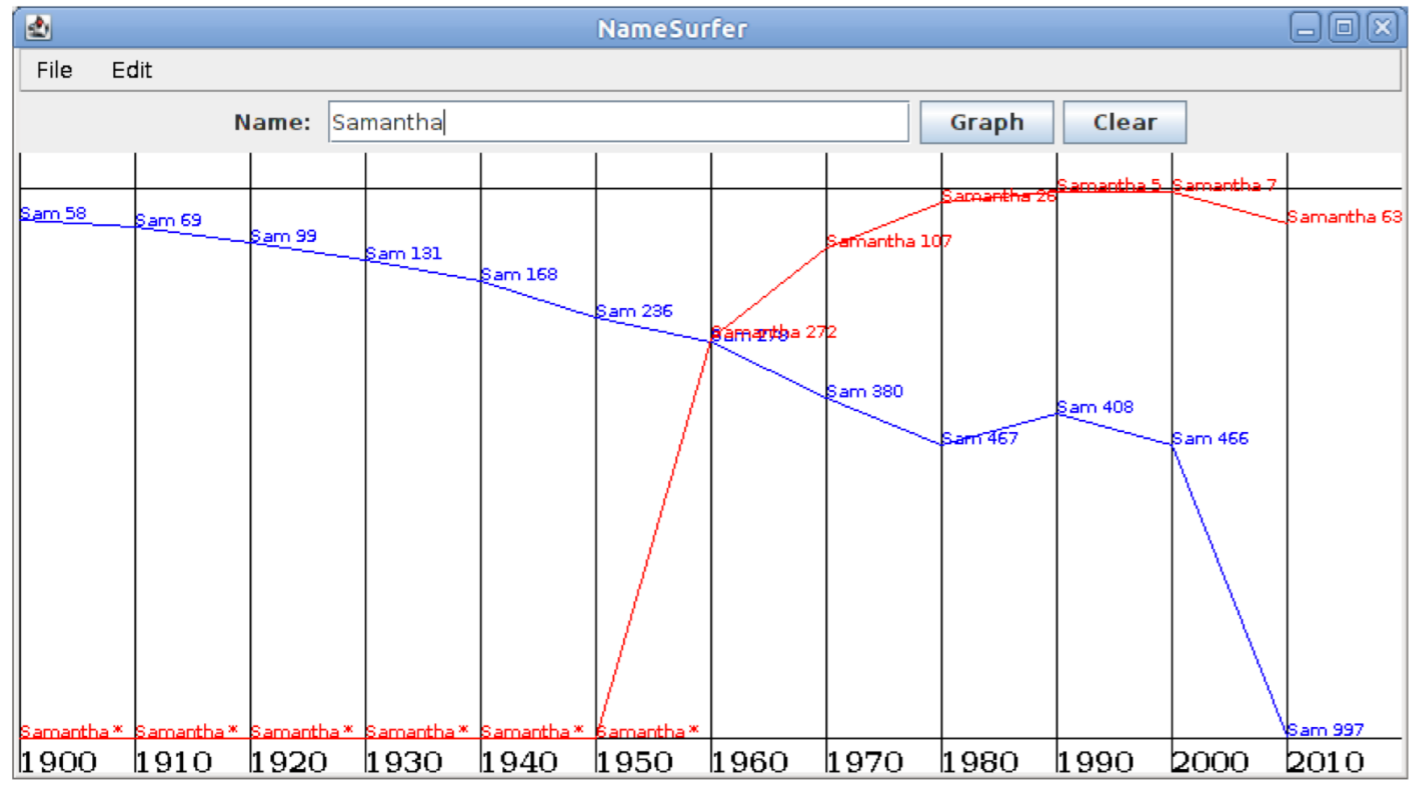
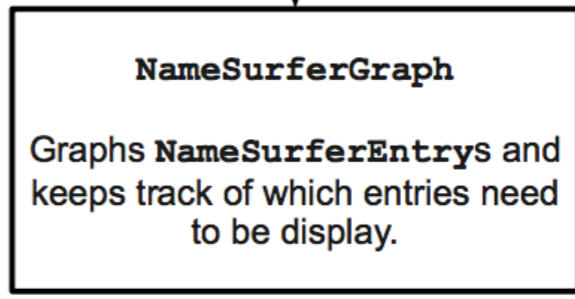


Asks for NameSurferEntry



Returns NameSurferEntry

Hands over NameSurferEntry that need to be displayed.



Tricky parts

Tricky parts

- null pointer exceptions

Tricky parts

- null pointer exceptions
- `OutOfBoundsException`

Tricky parts

- null pointer exceptions
- `OutOfBoundsException`
- off-by-one drawing

Tricky parts

- null pointer exceptions
- `OutOfBoundsException`
- off-by-one drawing
- resizing using `update()`