## Algorithm analysis, Bubble Sort

ArrayList <game> list = c.getWishList(); <math>c_1</math> for (int i = 0; i &lt; list.size(); i++) <math>c_2</math> <math>n+1</math> for (int j = 1; j &lt; list.size(); i++) <math>c_3</math> <math>\frac{n(n+1)}{2} + n</math> Shelve s1 = searchShelve(list.get(j - 1).getShelveName()); <math>c_4</math> <math>\frac{n(n+1)}{2}</math> Shelve s2 = searchShelve(list.get(j).getShelveName()); <math>c_5</math> <math>\frac{n(n+1)}{2}</math> if (s1.getNameShelve().compareTo(s2.getNameShelve()) == 0) <math>c_6</math> <math>\frac{n(n+1)}{2}</math> if (s1.getGameShelve().getIndexInTable(list.get(j - 1).getCode()) &gt; s1.getGameShelve() <math>c_7</math> <math>\frac{n(n+1)}{2}</math> Game temp1 = list.get(j - 1); <math>c_8</math> <math>\frac{n(n+1)}{2}</math> list.set(j - 1, list.get(j)); <math>c_9</math> <math>\frac{n(n+1)}{2}</math> list.set(j, temp1); <math>c_9</math> <math>\frac{n(n+1)}{2}</math> <math>\frac{n(n+1)}{2}</math> <math>\frac{n(n+1)}{2}</math> <math>\frac{n(n+1)}{2}</math> list.set(j - 1, list.get(j) - 1); <math>c_{10}</math> <math>\frac{n(n+1)}{2}</math> <math>\frac{n(n+1)}{2}</math> <math>\frac{n(n+1)}{2}</math> list.set(j - 1, list.get(j)); <math>c_{11}</math> <math>\frac{n(n+1)}{2}</math> <math>\frac{n(n+1)}{2}</math> <math>\frac{n(n+1)}{2}</math> <math>\frac{n(n+1)}{2}</math> <math>\frac{n(n+1)}{2}</math> list.set(j - 1, list.get(j)); <math>c_{12}</math> <math>\frac{n(n+1)}{2}</math> <math></math></game>	Instruction	Cost	# Repeat
	ArrayList <game> list = c.getWishList();</game>	$c_1$	1
$ \begin{array}{c ccccccccccccccccccccccccccccccccccc$	for (int i = 0; i < list.size(); i++)	$c_2$	n+1
$ \begin{array}{c ccccccccccccccccccccccccccccccccccc$	for (int $j = 1$ ; $j < list.size() - i$ ; $j++$ )	$c_3$	$\frac{n(n+1)}{2} + n$
	Shelve s1 = searchShelve(list.get(j - 1).getShelveName());	C <sub>4</sub>	$\frac{n(n+1)}{2}$
$ \begin{array}{c} & \hline \\ \text{if (s1.getGameShelve().getIndexInTable(list.get(j-1).getCode())} \\ \text{.getIndexInTable(list.get(j).getCode()))} \\ \hline \\ \text{Game temp1 = list.get(j-1);} \\ \\ \text{list.set(j-1, list.get(j));} \\ \hline \\ \text{else if (s1.getNameShelve().compareTo(s2.getNameShelve())} > 0) } \\ \hline \\ \text{Game temp2 = list.get(j-1);} \\ \hline \\ \text{Game temp2 = list.get(j-1);} \\ \hline \\ \text{list.set(j-1, list.get(j));} \\ \hline \\ \text{c}_{12} \\ \hline \\ \text{c}_{13} \\ \hline \\ \text{c}_{14} \\ \hline \\ \text{c}_{15} \\ \hline \\ \text{c}_{15$	Shelve s2 = searchShelve(list.get(j).getShelveName());	<i>c</i> <sub>5</sub>	$\frac{n(n+1)}{2}$
$ \begin{array}{c c} \\$	if (s1.getNameShelve().compareTo(s2.getNameShelve()) == 0)	<i>c</i> <sub>6</sub>	$\frac{n(n+1)}{2}$
$\begin{array}{c ccccc} \text{Game temp1} = \text{list.get(j-1)}; & & c_8 & \frac{n(n+1)}{2} \\ & \text{list.set(j-1, list.get(j))}; & & c_9 & \frac{n(n+1)}{2} \\ & \text{list.set(j, temp1)}; & & c_{10} & \frac{n(n+1)}{2} \\ & \text{else if (s1.getNameShelve().compareTo(s2.getNameShelve())} > 0) \left\{ & & c_{11} & \frac{n(n+1)}{2} \\ & & & c_{12} & \frac{n(n+1)}{2} \\ & & & & c_{13} & \frac{n(n+1)}{2} \\ & & & & c_{14} & \frac{n(n+1)}{2} \\ & & & $		<i>c</i> <sub>7</sub>	$\frac{n(n+1)}{2}$
		<i>c</i> <sub>8</sub>	$\frac{n(n+1)}{2}$
else if (s1.getNameShelve().compareTo(s2.getNameShelve()) > 0) { $ c_{11} = \frac{n(n+1)}{2} $ Game temp2 = list.get(j - 1); $ c_{12} = \frac{n(n+1)}{2} $ list.set(j - 1, list.get(j)); $ c_{13} = \frac{n(n+1)}{2} $ list.set(j, temp2); $ c_{14} = \frac{n(n+1)}{2} $	list.set(j - 1, list.get(j));	<i>C</i> <sub>9</sub>	$\frac{n(n+1)}{2}$
	list.set(j, temp1);	c <sub>10</sub>	$\frac{n(n+1)}{2}$
list.set(j - 1, list.get(j)); $ c_{13} = \frac{\frac{1}{2}}{\frac{n(n+1)}{2}} $ list.set(j, temp2); $ c_{14} = \frac{n(n+1)}{2} $	else if (s1.getNameShelve().compareTo(s2.getNameShelve()) > 0) {	c <sub>11</sub>	$\frac{n(n+1)}{2}$
list.set(j, temp2); $ c_{14} = \frac{\frac{1}{2}}{\frac{n(n+1)}{2}} $	Game temp2 = list.get(j - 1);	$c_{12}$	$\frac{n(n+1)}{2}$
	list.set(j - 1, list.get(j));	c <sub>13</sub>	$\frac{n(n+1)}{2}$
c setWishList(list):	list.set(j, temp2);	c <sub>14</sub>	$\frac{n(n+1)}{2}$
	c.setWishList(list);	c <sub>15</sub>	1

**Note:** We assume that n = list. size

## Analysis of time complexity

$$T(n) = c_1 + c_2(n+1) + c_3\left(\frac{n(n+1)}{2} + n\right) + c_4\left(\frac{n(n+1)}{2}\right) + c_5\left(\frac{n(n+1)}{2}\right) + c_6\left(\frac{n(n+1)}{2}\right) + c_7\left(\frac{n(n+1)}{2}\right) + c_8\left(\frac{n(n+1)}{2}\right) + c_8\left(\frac{n(n+1)}{2}\right) + c_{10}\left(\frac{n(n+1)}{2}\right) + c_{11}\left(\frac{n(n+1)}{2}\right) + c_{12}\left(\frac{n(n+1)}{2}\right) + c_{13}\left(\frac{n(n+1)}{2}\right) + c_{14}\left(\frac{n(n+1)}{2}\right) + c_{15}$$

$$T(n) = c_1 + c_2 + c_2 n + c_3 n + \frac{c_3 n^2 + c_3 n + c_4 n^2 + c_4 n + c_5 n^2 + c_5 n + c_6 n^2 + c_6 n + c_7 n^2 + c_7 n + c_8 n^2 + c_8 n + c_9 n^2 + c_9 n}{2} + \frac{c_{10} n^2 + c_{10} n + c_{11} n^2 + c_{11} n + c_{12} n^2 + c_{12} n + c_{13} n^2 + c_{13} n + c_{14} n^2 + c_{14} n}{2} + c_{15}$$

$$T(n) = n^2 \left( \frac{c_3}{2} + \frac{c_4}{2} + \frac{c_5}{2} + \frac{c_6}{2} + \frac{c_7}{2} + \frac{c_8}{2} + \frac{c_9}{2} + \frac{c_{10}}{2} + \frac{c_{11}}{2} + \frac{c_{12}}{2} + \frac{c_{13}}{2} + \frac{c_{14}}{2} \right) + n \left( c_2 + c_3 + \frac{c_3}{2} + \frac{c_4}{2} + \frac{c_5}{2} + \frac{c_6}{2} + \frac{c_7}{2} + \frac{c_8}{2} + \frac{c_9}{2} + \frac{c_{10}}{2} + \frac{c_{11}}{2} + \frac{c_{12}}{2} + \frac{c_{13}}{2} + \frac{c_{14}}{2} \right) + (c_1 + c_2 + c_{15})$$

Assuming that

$$k = \frac{c_3}{2} + \frac{c_4}{2} + \frac{c_5}{2} + \frac{c_6}{2} + \frac{c_7}{2} + \frac{c_8}{2} + \frac{c_9}{2} + \frac{c_{10}}{2} + \frac{c_{11}}{2} + \frac{c_{12}}{2} + \frac{c_{13}}{2} + \frac{c_{14}}{2}$$

$$y = c_2 + c_3 + \frac{c_3}{2} + \frac{c_4}{2} + \frac{c_5}{2} + \frac{c_6}{2} + \frac{c_7}{2} + \frac{c_8}{2} + \frac{c_9}{2} + \frac{c_{10}}{2} + \frac{c_{11}}{2} + \frac{c_{12}}{2} + \frac{c_{13}}{2} + \frac{c_{14}}{2}$$

$$z = c_1 + c_2 + c_{15}$$

## We have that:

$$T(n) = n^2(k) + n(y) + z \to \theta(n^2)$$

## Spatial complexity analysis

Type	Variable	Size of 1 atomic value	Amount values
Input	List	32 bits	1
Auxiliar	S1	32 bits	1
Auxiliar	S2	32 bits	1

Auxiliar	Temp1	32 bits	1
Auxiliar	Temp2	32 bits	1

- Total Spatial Complexity =  $Input + Auxiliary + Output = 5 = \theta(5)$
- Auxiliary Spatial Complexity =  $4 = \theta(4)$
- Auxiliary Spatial Complexity Output =  $Auxiliary + Output = 4 = \theta(4)$