|  |  |  |  |
| --- | --- | --- | --- |
| Iteration | psn | before | after |
| 1 | 9 | “ “ | “ “ |
| 1 | 0 | “” | “ “ |
| 1 | 8 | “ “ | “ c ” |
| 2 | 19 | “ “ | “s” |
| 1 | 3 | “k” | “w” |
| 2 | 7 | “ “ | “t” |
| 3 | 21 | “s” | “w” |

Magpie Activity 3: Better Keyword Detection

**private** String transformIWantToStatement(String statement)

{

// Remove the final period, if there is one

statement = statement.trim();

String lastChar = statement.substring(statement

.length() - 1);

**if** (lastChar.equals("."))

{

statement = statement.substring(0, statement

.length() - 1);

}

**int** psn = findKeyword (statement, "I want ", 0);

String restOfStatement = statement.substring(psn + 9).trim();

**return** "Would you really be happy if you " + restOfStatement + "?";

}

Magpie Activity 4:

**public** String getResponse(String statement)

{

String response = "";

**if** (statement.length() == 0)

{

response = "Say something, please.";

}

**else** **if** (findKeyword(statement, "no") >= 0)

{

response = "Why so negative?";

}

**else** **if** (findKeyword(statement, "mother") >= 0

|| findKeyword(statement, "father") >= 0

|| findKeyword(statement, "sister") >= 0

|| findKeyword(statement, "brother") >= 0)

{

response = "Tell me more about your family.";

}

// Responses which require transformations

**else** **if** (findKeyword(statement, "I want", 0) >= 0)

{

response = transformIWantToStatement(statement);

}

**else** **if** (findKeyword(statement, "I", 0) >=0 && findKeyword(statement, "you", 0) >=0 )

{

response = transformYouMeStatement(statement);

}

**else**

{

// Look for a two word (you <something> me)

// pattern

**int** psn = findKeyword(statement, "you", 0);

**if** (psn >= 0

&& findKeyword(statement, "me", psn) >= 0)

{

response = transformYouMeStatement(statement);

}

**else**

{

response = getRandomResponse();

}

}

**return** response;

}

/\*\*

\* Take a statement with "I want to <something>." and transform it into

\* "What would it mean to <something>?"

\* **@param** statement the user statement, assumed to contain "I want to"

\* **@return** the transformed statement

\*/

**private** String transformIWantToStatement(String statement)

{

// Remove the final period, if there is one

statement = statement.trim();

String lastChar = statement.substring(statement

.length() - 1);

**if** (lastChar.equals("."))

{

statement = statement.substring(0, statement

.length() - 1);

}

**int** psn = findKeyword (statement, "I want ", 0);

String restOfStatement = statement.substring(psn + 8).trim();

**return** "Would you really be happy if you had " + restOfStatement + "?";

}

/\*\*

\* Take a statement with "you <something> me" and transform it into

\* "What makes you think that I <something> you?"

\* **@param** statement the user statement, assumed to contain "you" followed by "me"

\* **@return** the transformed statement

\*/

**private** String transformYouMeStatement(String statement)

{

// Remove the final period, if there is one

statement = statement.trim();

String lastChar = statement.substring(statement

.length() - 1);

**if** (lastChar.equals("."))

{

statement = statement.substring(0, statement

.length() - 1);

}

**int** psnOfMe = findKeyword (statement, "you", 5);

String restOfStatement = statement.substring(2, psnOfMe);

**return** "Why do you " + restOfStatement + " me?";

}

Magpie Activity 5:

**private** String getRandomResponse ()

{

Random r = **new** Random ();

**return** randomResponses [r.nextInt(randomResponses.length)];

}

**private** String [] randomResponses = {"Interesting, tell me more",

"Hmmm.",

"Do you really think so?",

"You don't say.",

"I don't like your awnser","Thats prettty sweet tell me more...", "what did you say?", "this bores me, say something else.."

};

}// end of method