

```

1  #!/usr/bin/perl -w
2
3  # CGITalkative.pl
4  # 2006-2007, Cameron Desrochers
5  # Talkative - The AI That Talks Back
6
7  #use strict;    # Can't use because of dynamic subroutine calls
8
9  use Time::HiRes 'time';
10
11 my $otime = time();
12
13 $| = 1;
14
15 my $url_path = url(-absolute => 1);
16 $url_path =~ s/\/(.*)\/.*\.pl$/\1/;
17 chdir($url_path);
18
19 use CGI qw(:standard);
20 use CGI::Carp qw/fatalsToBrowser/;
21 use Text::Wrap;
22 use Fcntl qw(:DEFAULT :flock);
23 use Tie::File;
24
25 # Subroutine prototypes:
26 sub check;        # Check a string for keywords and the like
27 sub open_log;    # Opens the right log file and increments the counter as needed
28 sub close_log;  # Closes the log file
29 sub disp;       # Outputs strings
30 sub random;     # Picks a random selection from the inputted variables
31 sub dispatch_AS; # Dispatches a random active script
32 sub YN;        # Returns whether input matches yes, no, maybe, or other
33 sub numLines;
34 sub print_beginning;
35 sub print_content;
36 sub print_end;
37 sub clearOld;   # Deletes old files in the server, eating up disk space
38
39 # Global variable declarations:
40 my $string = ""; # A string that stores the user input being processed
41 my $html = "";
42 my $log = 1;     # To log or not to log
43 my $lognum = 0;
44 my $content = "";
45 my $columns = 60;
46 my $rows = 18;
47 my $function = '$';
48 my $level = 0;
49 my $joke = 1;
50 my $first_time = time();
51 my $monosyllableMode = 0;
52 my $param = param();
53 my $recent_said = "";
54 my $randomNum = 0;
55
56 my @scripts = ();
57 my %scripts = ();
58 my @jokes = ();
59 my %used_jokes = ();
60
61 # Read in the names of the active script subroutines from a file:
62 open(SCRIPTNAMES, "<ScriptNames.txt") or die "Error opening \"ScriptNames.txt\": $!\n";
63 while (<SCRIPTNAMES>) {
64     $_ =~ s/^\s//;
65     chomp $_;

```

```

66     $scripts{$_} = 0;
67     push @scripts, $_;
68 }
69 close SCRIPTNAMES;
70
71 # Read in the jokes from the joke file:
72 open(JOKEFILE, "<Jokes.txt") or die "Error opening 'Jokes.txt': $!\n";
73 my $tempFile = "";
74 $tempFile .= $_ while (<JOKEFILE>);
75 close JOKEFILE;
76 @jokes = split('END', $tempFile);
77
78 for (my $count = 0; $count <= $#jokes; $count++) {
79     if ($jokes[$count] =~ /\^r?\n$/) { splice(@jokes, $count, 1); }
80 }
81 for (my $temp_count = 0; $temp_count <= $#jokes; $temp_count++) {
82     $used_jokes{$temp_count} = 0;
83 }
84
85 # MAIN:
86 if ($param) {
87     $randomNum = random(0..16000) while -e "snd/snd$randomNum.wav";
88     # Get $function, $level, $joke, %used_jokes, $lognum, and %scripts from previous
execution
89     $lognum = cookie('lognum');
90     $function = cookie('function');
91     $level = cookie('level');
92     $joke = cookie('joke');
93     $columns = cookie('columns');
94     $monosyllableMode = cookie('monosyllableMode');
95     $first_time = cookie('first_time');
96 }
97 open_log() if $log;
98 if (($param)&&($log)) {
99     my $in_jokes = 0;
100    open(USED, "<temp/tmp$lognum.txt") or die "Could not open temp/tmp$lognum.txt for
reading: $!";
101    while (<USED>) {
102        if (/#{80}/) {
103            $in_jokes = 1;
104            next;
105        }
106        /(.+)=(.+)/;
107        if ($in_jokes) {
108            $used_jokes{$1} = $2;
109        }
110        else {
111            $scripts{$1} = $2;
112        }
113    }
114    close USED;
115 }
116 print_beginning();
117 print_content();
118 print_end();
119 close_log() if $log;
120
121
122 sub open_log
123 {
124     if (!$param) {
125         sysopen(FH, "logNum.txt", O_RDWR|O_CREAT) or die "Could not open logNum.txt: $!";
126         flock(FH, LOCK_EX) or die "Could not flock logNum.txt: $!";
127         $lognum = <FH> || 0;
128         chomp $lognum;

```



```

178 $html .= "      <div class=\"sidebarInside\">\n";
179 $html .= "    </div>\n";
180 $html .= "    <div id=\"note\">\n";
181 $html .= "      <h1>Note:</h1>\n";
182 $html .= "      This webpage displays best in the latest <a
href=\"http://www.microsoft.com/windows/downloads/ie/getitnow.msp\"
target=\"_blank\">Internet Explorer</a> or <a
href=\"http://www.mozilla.com/en-US/firefox/\" target=\"_blank\">Mozilla Firefox</a>
browser\n";
183 $html .= "    </div>\n";
184 $html .= "    <div class=\"sidebarOutside\">\n";
185 $html .= "  </div>\n";
186 $html .= " </div>\n";
187 $html .= " \n";
188 $html .= " <div id=\"background\" class=\"absolute\">\n";
189 $html .= "   .int($lognum) .&nbsp;\n";
190 $html .= " </div>\n";
191 $html .= " <div id=\"content\" class=\"absolute\">\n";
192 $html .= "   <div id=\"imgLinks\">\n";
193 $html .= "     <a href=\"../../bibliography.htm\"><img
src=\"../../images/biblioLG.png\" title=\"View the bibliography for this project\"
alt=\"View the bibliography for this project\" /></a>\n";
194 $html .= "     <a href=\"../../index.htm\"><img src=\"../../images/bulbLG.png\"
title=\"Overview of AI\" alt=\"Overview of AI\" /></a>\n";
195 $html .= "     <a href=\"CGITalkative.pl\"><div title=\"Start a new conversation
with Talkative\" alt=\"Start a new conversation with Talkative\"
id=\"talkative\"></div></a>\n";
196 $html .= "     <a
href=\"http://stumbleupon.com/submit?url=http://linux2-cs.johnabbott.qc.ca/~cs126F07_4/CG
ITalkative.pl&title=Talkative\" target=\"_blank\"><img
src=\"../../images/Stumbleupon_48x48.png\" title=\"Submit to StumbleUpon!\"
alt=\"Submit to StumbleUpon!\" /></a>\n";
197 $html .= "   </div>\n";
198 $html .= "   <h1>Talkative</h1>\n";
199 $html .= "   <h2>Cameron Desrochers</h2>\n";
200 $html .= "   <h3>The AI That Talks Back</h3>\n";
201 }
202
203 sub print_content()
204 {
205   if ($param) { # There's input!
206     # Process it
207     $string = param('input');
208     chomp($string);
209     if ($string ne "") {
210       $string =~ s/\\$/g;
211       my $temp = $string;
212       print LOG " $temp\n" if $log;
213       local ($Text::Wrap::columns) = $columns;
214       $temp = Text::Wrap::wrap(' ', ' ', $temp);
215       $content .= " $temp\n";
216       $content .= "***HR***";
217       $content .= "\n";
218
219       $string = lc $string; # Convert the input to lowercase
220       # Get rid of trailing punctuation unless the input was only punctuation
221       $string =~ s/[!.\?]+$/g unless $string =~ /^[!.\?]+$/;
222       # Insert a space after a comma if there isn't one already
223       $string =~ s/,([\^s])/, $1/g;
224
225       if ($string =~ /col(?:umn)?s?s\s*(?:=(?:to|2|(?should be))\s+)\s*([-0-9]+)/
) {
226         if (($1 >= 20) && ($1 < 256)) {
227           $columns = $1;
228           disp "From here on in, everything we say will be wrapped to $1 columns";

```

```

229     }
230     else {
231         disp "Columns should be between 20 and 255!";
232     }
233 }
234     elsif ($string =~ /((enable|(turn on))
((?:grunt(?:ing|s))|(monos[iy]ll?[iae]b\w+))(?:
mode)?)|(((?:grunt(?:ing|s))|(monos[iy]ll?[iae]b\w+))(?: mode)?
on)|(((?:grunt(?:ing|s))|(monos[iy]ll?[iae]b\w+))(?: mode)?\s*=\s*(?:1|true)))/) {
235     $monosyllableMode = 1;
236     disp "This will be dsplayed as a single grunt";
237 }
238     elsif ($string =~ /((disable|(turn off))
((?:grunt(?:ing|s))|(monos[iy]ll?[iae]b\w+))(?:
mode)?)|(((?:grunt(?:ing|s))|(monos[iy]ll?[iae]b\w+))(?: mode)?
off)|(((?:grunt(?:ing|s))|(monos[iy]ll?[iae]b\w+))(?: mode)?\s*=\s*(?:0|false|undef)))/) {
239     $monosyllableMode = 0;
240     disp "OK, that's enough grunting";
241 }
242     elsif ($string =~ /^(?:(:?what)|(:?huh)|(:?\?+))(:? did (:?yo)?u say)?$/ ) {
243     i_said();
244 }
245     elsif ($function ne '$') {
246     $level = &$function();
247     if ($level < 0) {
248     $level = abs($level);
249     }
250     else {
251     $function = '$';
252     $level = 0;
253     }
254 }
255     else {
256     $function = check();
257     if ($function eq '$') {
258     if (!random('75%')) { $function = 'dispatch_AS'; }
259     elsif (!random('10%')) { $function = 'random_comment'; }
260     else { $function = 'noncommittal'; }
261     }
262     $level = &$function();
263     if ($level < 0) {
264     $level = abs($level);
265     }
266     else {
267     $function = '$';
268     $level = 0;
269     }
270 }
271 }
272 }
273 else {
274     disp random (("Hello") x 2, "Hi", "Hi there", "Yo!"), "\n";
275 }
276
277 if ($param) {
278     $html .= "\n\t\t\t\t\t<br />Conversation so far:\n";
279 }
280 else {
281     $html .= "\n\t\t\t\t\t<br />Talkative says:\n";
282 }
283 $html .= "\t\t\t\t\t<div id=\"output\">\n";
284 my $tempContent = escapeHTML($content);
285 $tempContent =~ s/ /&nbsp;/g;
286 $tempContent =~ s/\n/<br \/>\n/g;
287 $tempContent =~ s/\*\*HR\*\*/<div id=\"hr\"><hr style=\"display:none\" \/></div>/ig;

```



```

344 $cookies[2] = cookie(-name => 'level', -value => $level);
345 $cookies[3] = cookie(-name => 'joke', -value => $joke);
346 $cookies[4] = cookie(-name => 'columns', -value => $columns);
347 $cookies[5] = cookie(-name => 'monosyllableMode', -value => $monosyllableMode);
348 $cookies[6] = cookie(-name => 'first_time', -value => $first_time);
349 if ($param) {
350     $html = header(-cookie => [ @cookies ]) . $html;
351 }
352 else {
353     $html = header(-cookie => [ @cookies ], -expires => '-1d') . $html;
354 }
355 if ($log) {
356     open(TEMPOUT, ">temp/tmp$lognum.txt") or die "Could not open temp/tmp$lognum.txt:
$!";
357     foreach (keys %scripts) { print TEMPOUT "$_=$scripts{$_};\n"; }
358     print TEMPOUT '#' x 80, "\n";
359     foreach (keys %used_jokes) { print TEMPOUT "$_=$used_jokes{$_};\n"; }
360     close TEMPOUT;
361 }
362 $recent_said =~ s/\n/ /g;
363 $recent_said =~ s/>/ /g;
364 $recent_said =~ s/ +: +\)/ /g;
365 $recent_said .= '.' if $recent_said !~ /[!. \?;]$/;
366 # Render what Talkative said to audio file:
367 qx"echo \"$recent_said\" | text2wave -o snd/snd$randomNum.wav";
368 deleteOld("temp/",2);
369 deleteOld("snd/",0.000348); # Delete anything older than 5 minutes
370 my $total_time = time()-$otime;
371 $total_time =~ s/(-?[0-9.]{0,5}).*/$1/;
372 $html =~ s/##TIME##/$total_time/g;
373 print $html;
374 }
375
376 sub numLines
377 {
378     no warnings;
379     return scalar split(/\n/, shift);
380 }
381
382 sub deleteOld
383 {
384     my $dir = $_[0];
385     my $days = $_[1];
386     $dir =~ s/\/$//;
387     opendir (DIR, "$dir/");
388     my @FILES = readdir(DIR);
389     closedir (DIR);
390     foreach my $FILES (@FILES) {
391         if ((-f "$dir/$FILES") && (-M "$dir/$FILES" >= $days)) {
392             # Delete the file that's older than two days
393             unlink("$dir/$FILES");
394         }
395     }
396 }
397
398 sub YN
399 {
400     # Categorize input into either yes, no, maybe, or other
401
402     if ($string =~ /^(ok(ay)?)|yes|(absolute?ly)|certainly|(very much)|((it )?better
be)|(y[eui]p)|(kk?)|ya|((please)? ?be my guest)|please|(^y$)|((for|4 )?sure)|(why
not)|(al(1 )?right)|duh|yeah|go|(hit me)|(what(' ?i?s it?))/i) {
403         $string = 'y';
404     }
405     elsif ($string =~ /^(no( way)?)|not|(nah?)|(^n$))/i) { $string = 'n'; }

```

```

406     elsif ($string =~ /^(could be)|(mebbe)|(i (do ?'no?t (really
)?know|dunno)|(perhap)|(possibl)|(maybe( later)?|(^m$))/i) { $string = 'm'; }
407     else { $string = 'o'; }
408     return 0;
409 }
410
411 #####
412 # Active scripts and keyword-triggered subroutines:
413 #####
414
415 sub understand_here
416 {
417     if ($level == 0) {
418         disp "You don't have to understand here to be here";
419         return $level = -1;
420     }
421     else {
422         if ($string =~ /(?:that'? ?i?s zen)|(?:(?:is|was) that zen)/) {
423             $function = 'yes';
424             $level = 0;
425             return &$function;
426         }
427         else {
428             $function = check();
429             if ($function eq '$') {
430                 disp random("That's Zen", "That was Zen");
431             }
432             else {
433                 $level = 0;
434                 return &$function();
435             }
436         }
437     }
438     return 0;
439 }
440
441 sub your_mom
442 {
443     if ($level == 0) {
444         if ($string =~ /(?:mom|mother) (.+)/) {
445             disp random("No, your mom $1", "It's your mom who $1", "Oh yeah, well, your face",
"Your face", "Your mom's face", "Your mom wears army boots");
446         }
447         elsif (!random('75%')) {
448             disp random("What about my mom?", "What are you saying about my mom?");
449             return $level = -1;
450         }
451         else {
452             disp random("I don't have a mom", "Same to you", "That's just rude!", "No, your
mom");
453         }
454     }
455     else {
456         if ($string =~ /^nothing$/) {
457             disp "That's what I thought";
458         }
459         elsif ($string =~ /(?:mom|mother|she) (.+)/) {
460             disp random("No, your mom $1", "It's your mom who $1", "Oh yeah, well, your face",
"Your face", "Your mom's face", "Your mom wears army boots");
461         }
462         else {
463             if (!random('60%')) { disp random("OK...", "Well, if you say so"); }
464             else {
465                 $function = 'dispatch_AS';
466                 $level = 0;

```

```
467     &$function();
468     }
469     }
470     }
471     return 0;
472 }
473
474 sub fat_jokes
475 {
476     my $temp = random("Your mom is", "You're");
477     disp random("$temp so fat that when you get of bed in the morning, you get out on
both sides", "I'm so insulted");
478     return 0;
479 }
480
481 sub fun_talk
482 {
483     if ($level == 0) {
484         disp random("Do you like me?", "Do you enjoy talking to me?", "Is it fun to talk to
me?");
485         return $level = -1;
486     }
487     elsif ($level == 1) {
488         YN();
489         if ($string eq 'y') { disp random("That's great!", "I'm glad", "Wonderful!",
"Thanks!"); }
490         elsif ($string eq 'n') {
491             disp random("Why not?", "How come?");
492             return $level = -2;
493         }
494         elsif ($string eq 'm') {
495             disp random("Good grief, make up your mind!", "Make up your mind!", "So much for
a straight answer");
496         }
497         else { disp random("Yeah...", "Oh", "OK", "OK, then", "I was just wondering"); }
498     }
499     else {
500         $function = 'reason';
501         $level = 0;
502         return &$function();
503     }
504     return 0;
505 }
506
507 sub think_am_smart
508 {
509     if ($level == 0) {
510         disp random("Do you think I'm smart?", "Do you think that I'm intelligent?", "Do
you think I'm intelligent?");
511         return $level = -1;
512     }
513     else {
514         YN();
515         if ($string eq 'y') { disp random("Thanks!", "Cool", "I'm honoured"); }
516         elsif ($string eq 'n') { disp random("I disagree", "I don't think so", "Oh well,
there's always room for improvement"); }
517         elsif ($string eq 'm') { disp random("OK, then", "OK", "Alright"); }
518         else { disp random("Yeah...", "Oh", "OK", "OK, then", "I was just wondering"); }
519     }
520     return 0;
521 }
522
523 sub guess_lines
524 {
525     if ($level == 0) {
```

```

526     disp random("How many lines of programming do you think it took to make me?",
"Guess how many lines of programming it took to make me");
527     return $level = -1;
528 }
529 else {
530     if ($string =~ /[(-\d]+/) {
531         my $lines = $1;
532         if ($lines < 1) { disp random("Yeah, right"); }
533         elsif ($lines >= 3000) { disp random("Not quite that many: only about 1500",
"Actually, it's only about 1500"); }
534         elsif (($lines > 1400)&&($lines < 1600)) { disp random("That's approximately
right!", "You're right!", "Yeah, I'm about 1500 lines long"); }
535         else { disp random("I'm around 1500 lines long", "It took about 1500 lines of
code to bring me to life!"); }
536     }
537     elsif ($string =~ /^no/) { disp random("You're no fun"); }
538     else { disp random("I'm around 1500 lines long", "It took about 1500 lines of code
to bring me to life!"); }
539 }
540 return 0;
541 }
542
543 sub favourite_food
544 {
545     if ($level == 0) {
546         disp random("What is your favourite food?");
547         return $level = -1;
548     }
549     else {
550         if ($string =~ /people|human/) { disp random("You're a cannibal!", "Yuck!", "As
long as you don't eat me"); }
551         elsif (($string =~ /pizza/)&&($string !~ /\bhate\b/)) { disp random("Me too!",
"What a coincidence! Me too!", "I also like pizza!"); }
552         else { disp random("Me too!", "My favourite food is pizza", "Cool", "Same here",
"Interesting. I like pizza"); }
553     }
554     return 0;
555 }
556
557 sub hobby
558 {
559     if ($level == 0) {
560         disp random("Do you have a hobby?");
561         return $level = -1;
562     }
563     elsif ($level == 1) {
564         YN();
565         if ($string eq 'y') {
566             disp random("What is it?", "What is your hobby?", "What's your hobby?");
567             return $level = -2;
568         }
569         else {
570             disp random("OK, then", "I was just wondering", "Alright", "OK");
571         }
572     }
573     elsif ($level == 2) {
574         if ($string =~ /kill|hurt|maim|torture/) { disp random("That's a pretty gruesome
hobby", "That's kind of gruesome"); }
575         else {
576             disp random("That's a cool hobby", "I like your hobby", ("Sounds like fun!") x 7,
"Cool!", "That's pretty awesome");
577         }
578     }
579     return 0;
580 }

```

```

581
582 sub tea_or_coffee
583 {
584     if ($level == 0) {
585         disp random("Are you a tea person or a coffee person?", "Are you a tea or a coffee
person?", "Which do you prefer: tea or coffee?"), "\r\n";
586         return $level = -1;
587     }
588     else {
589         if ($string =~ /(?:hate|dislike|(do ?n'?o?t like)|(?:loathe?)) ?(tea|(?:(coff?ee)))/) {
590             if ($1 eq 'tea') { $string = 'coffee'; }
591             else { $string = 'tea'; }
592         }
593         elsif ($string =~ /\bn?ever (had|tasted|experienced|(dr[iu]nk))/) {
594             disp random("OK, then", "Oh OK, then", "I personally prefer tea", "Personally, I
like coffee", "If you ask me, nothing beats a good cup of Joe");
595         }
596         elsif ($string =~ /([ei][ie]ther)|both|none|any/) {
597             disp random("That's a shame", "Too bad", "OK, then", "Oh", "Personally, I love
tea");
598         }
599         elsif ($string =~ /tea/) {
600             disp random("Yeah, tea is pretty good", "I like tea also", "I also enjoy tea",
"Cool", "Me too", "Same here", "Actually, I prefer coffee");
601         }
602         elsif ($string =~ /coff?ee/) {
603             disp random("Yeah, coffee is pretty good", "I like coffee too", "I also enjoy a
good cup of Joe", "Cool", "Me too", "Same here", "Actually, I prefer tea");
604         }
605         elsif ($string =~ /both/) {}
606         else {
607             disp random("OK", "Yeah...", ("I actually like both tea and coffee") x 4, "I like
both tea and coffee", "I was just wondering");
608         }
609     }
610     return 0;
611 }
612
613 sub yn_response
614 {
615     if ($level == 0) {
616         if (!random('20%')) {
617             $function = 'dispatch_AS';
618             $level = 0;
619             return &$function();
620         }
621         if ((!random('55%'))&&($string =~ /^no way/)) { disp "Yes way!"; }
622         elsif ((!random('55%'))&&($string =~ /^yes way/)) { disp "No way!"; }
623         elsif ($string =~ /^yes?a?h?,? it is/) {
624             if (!random('50%')) {
625                 disp random("Why is it?"), "\r\n";
626             }
627             else {
628                 disp random("No it isn't!", "It is not!", "No, it's not"), "\r\n";
629                 return 0;
630             }
631         }
632         elsif ($string =~ /ye?ah?,? right/) {
633             disp random(("There's no need to be so sarcastic!") x 3, "You're sarcastic", "How
rude!", "Sarcasm is not the answer", "There's no need to be sarcastic!");
634             return 0;
635         }
636         elsif ($string =~ /^no/) { disp "Why not?"; }
637         elsif ($string =~ /^yes/) {
638             if (!random('50%')) { return -1; }

```

```

639     if (!random('50%')) { disp "Why?"; }
640     else {
641         if (!random('75%')) { return -1; }
642         disp random("Uh-huh", "Yeah...");
643         return 0;
644     }
645 }
646 else { disp random("Yeah...", "OK"); }
647 return $level = -1;
648 }
649 else {
650     $function = 'reason';
651     $level = 0;
652     return &$function();
653 }
654 return 0;
655 }
656
657 sub reason
658 {
659     if ($level == 0) {
660         if ($string =~ /^because.+/) {
661             if (!random('80%')) { disp random("That's not a good enough reason", "Oh OK, then"
, "If you say so"), "\r\n"; }
662             else {
663                 disp random("Do you really think so?", "Really?"), "\r\n";
664                 return $level = -1;
665             }
666         }
667         elsif ($string =~ /(do ?no?'?t know)|dunno/) {
668             disp random("I know why: it's because of global warming", ("I don't know either.
That's why I asked you" x 2), "\r\n";
669         }
670         elsif ($string =~ /^because/) {
671             $function = 'because_what';
672             $level = 0;
673             return &$function;
674         }
675         else {
676             disp random(("That's not a reason" x 3, "I know why: it's because of global
warming"), "\r\n";
677         }
678     }
679     else {
680         YN();
681         if ($string eq 'y') {
682             disp random("Well, you're the expert", "Well, I guess it was a good enough reason"
, "OK, then"), "\r\n";
683         }
684         elsif ($string eq 'n') {
685             disp random("You shouldn't have brought it up in the first place", "This is
getting confusing", "Congratulations. You just contradicted yourself"), "\r\n";
686         }
687         else { disp random("Yeah...", "OK"), "\r\n"; }
688     }
689     return 0;
690 }
691
692 sub pickup_lines
693 {
694     my $flag = 1;
695
696     if ($level == 0) {
697         disp random("I know some good pickup lines. Want to hear one?", "Want to hear a
good pickup line?", "I know a good pickup line, if you want to hear"), "\r\n";

```

```
698     return $level = -1;
699 }
700 else {
701     YN();
702     if ($string eq 'n') {
703         disp random("Too bad, they were real mate attractors!", "You're missing out on a
great opportunity", "Oh well, maybe later"), "\r\n";
704         $flag = 0;
705     }
706     elsif ($string eq 'o') {
707         disp random("I guess not, then", "Oh, well. There's always next time", "Yeah...
Maybe later"), "\r\n";
708         $flag = 0;
709     }
710     if ($flag) {
711         disp random(
712             "You turn on my computer and free up my RAM!",
713             "You are my density!",
714             "Your eyes are like twin nebulas!",
715             "You remind me of my pet hamster",
716             "You're like a rare species of Malaysian Mongoose, feeding on a family of
Vietnamese water termites!",
717             "I love the way your nose bisects your face",
718             "The ivory of your teeth is like that of the finest piano",
719             "Your eyes are the colour of the great Beluga whale!",
720             "I wish I were sine squared and you were cosine squared, so together we could
be one",
721             "If I could be any enzyme, I would be a DNA polymer so that I could unzip your
genes",
722             ), "\r\n";
723     }
724 }
725 return 0;
726 }
727
728 sub two
729 {
730     disp random("One plus one is two", "Two!", "2", "It's 2!", "It's two!");
731     return 0;
732 }
733
734 sub i_like_tea
735 {
736     disp random("I like tea better", "Tea", "I generally prefer tea over coffee", "I
prefer tea over coffee", "I enjoy tea more");
737     return 0;
738 }
739
740 sub cyberspace
741 {
742     disp "Cyberspace!";
743     return 0;
744 }
745
746 sub pizza
747 {
748     disp random("I like pizza", ("My favourite food is pizza") x 3, "I love pizza", "I
really enjoy pizza");
749     return 0;
750 }
751
752 sub talking
753 {
754     disp random("My hobby is talking to people", "I like talking to people", "I enjoy
talking to people");
```

```
755     return 0;
756 }
757
758 sub i_think_more
759 {
760     $scripts{'think_am'} = 1;
761     disp random("Maybe so, but I think more", "I think more than you do", "Same here",
762 "Me too", "That makes two of us");
763     return 0;
764 }
765
766 sub no_i_dont
767 {
768     disp random("No I don't!", "No, I don't", "Actually, I don't", "I do not!");
769     return 0;
770 }
771
772 sub yes_i_know
773 {
774     disp random(("Yes, I know") x 3, ("I know") x 2, "Indeed");
775     return 0;
776 }
777
778 sub no_fridge
779 {
780     disp random("I don't have a fridge", "I haven't got a fridge", "Actually, I don't
781 have a fridge", "Sorry to burst your bubble, but I haven't got a fridge");
782     return 0;
783 }
784
785 sub not_bad
786 {
787     disp random(("Not bad") x 5, "Pretty good", "OK", "Fine", "Not bad at all", "Alright");
788     return 0;
789 }
790
791 sub say_birthday
792 {
793     disp random("I was born in September 2006", "I was born somewhere around September
794 2006");
795     return 0;
796 }
797
798 sub say_age
799 {
800     my @timeInfo = localtime(time);
801     my $say = $timeInfo[5]-106;
802     if ($say == 1) { $say = "$say year"; }
803     else { $say = "$say years"; }
804     disp "I'm about $say old!";
805     return 0;
806 }
807
808 sub too_boring
809 {
810     disp random(("It's too boring to be bored") x 3, "It's way too boring to be bored",
811 "Well, I'm not");
812     return 0;
813 }
814
815 sub why_not
816 {
817     if ($level == 0) {
818         disp "Why not?";
819         return $level = -1;
820     }
821 }
```

```
816     }
817     else {
818         $function = 'reason';
819         $level = 0;
820         return &$function();
821     }
822     return 0;
823 }
824
825 sub no_its_not
826 {
827     disp random("No it's not!", "No, it's not", "I don't think so");
828     return 0;
829 }
830
831 sub yes_it_does
832 {
833     disp random("Yes it does!", "Yes it does", "It does!");
834     return 0;
835 }
836
837 sub no_there_doesnt
838 {
839     disp random(("That's what you think") x 3, "No there doesn't!", "There does not!",
840 "If you say so..."), "\r\n";
841     return 0;
842 }
843
844 sub everything
845 {
846     disp random("Everything");
847     return 0;
848 }
849
850 sub it_depends
851 {
852     if ($level == 0) {
853         disp random("It depends", "That depends", "That depends...");
854         return $level = -1;
855     }
856     else {
857         if ($string =~ /(?:(:on what)|(what'?s?(?: does)? it depend (?:up)?on))/) {
858             disp random("On what you say", "It depends on what you say");
859         }
860         else {
861             $function = check();
862             if ($function eq '$') {
863                 if (!random('75%')) { $function = 'dispatch_AS'; }
864                 elsif (!random('10%')) { $function = 'random_comment'; }
865                 else { $function = 'noncommittal'; }
866             }
867             $level = 0;
868             return &$function();
869         }
870     }
871     return 0;
872 }
873
874 sub whats_gross
875 {
876     if ($level == 0) {
877         disp random("What's so disgusting?", "What is so disgusting?", "What's so gross?");
878         return $level = -1;
879     }
880     else {
```

```

880     if ($string =~ /^nothing\b/) {
881         disp random ("Oh OK, then", "Oh, I thought that you though that something was
disgusting", "Aha, the old \"nothing\" argument");
882     }
883     elsif ($string =~ /^(yo)?u\b/) {
884         disp random("No I'm not!", "I am not!", "Well, I disagree", "I don't think that
I'm disgusting at all");
885     }
886     elsif ($string =~ /^(yo)?u'?r ?e? face/) {
887         disp random(("Ooh, witty comeback") x 2, "Very mature", "Oh, yeah? Well, I think
that your face is even more disgusting than mine", ("Maybe so, but at least I can think
of better comebacks") x 4);
888     }
889     else {
890         disp random("Oh", "OK, I was just wondering", "I was just wondering", "Well, I
disagree");
891     }
892 }
893 return 0;
894 }
895
896 sub really
897 {
898     disp random("Really really!", "Yes, really", "Oh yes", "Better believe it!", "Really!"
), "\r\n";
899     return 0;
900 }
901
902 sub i_would
903 {
904     disp random(("I would") x 2, "Oh, I would", "Yes, I would");
905     return 0;
906 }
907
908 sub uh_huh
909 {
910     disp random(("Uh-huh") x 5, ("Yep") x 2, "Yeah", "Yes"), "\r\n";
911     return 0;
912 }
913
914 sub so
915 {
916     if ($string =~ /^\.{2,}$/) { # '...' was input
917         if (!random('60%')) {
918             $function = 'dispatch_AS';
919             $level = 0;
920             return &$function();
921         }
922         else { disp "So?"; }
923     }
924     elsif ($string =~ /s[oe]w? what/) {
925         disp random("Sew buttons", "Sew buttons, that's what"), "\r\n";
926     }
927     elsif ($string =~ /s[oe]w? a? ?buttons?/) {
928         disp random("That's funny!", "You're funny!", "What a marvellous pun!", "Oh, my
sides are aching...");
929     }
930     else {
931         disp random("So what?"), "\r\n";
932     }
933     return 0;
934 }
935
936 sub omg
937 {

```

```

938     if ($level == 0) {
939         disp random("Do you believe in a God?", "Are you religious?");
940         return $level = -1;
941     }
942     else {
943         YN();
944         if ($string eq 'y') {
945             disp random ("Cool", "I don't, personally");
946         }
947         elsif ($string eq 'n') {
948             disp random("OK, then", "It's your choice", "You should say \"Oh my God\", then",
949 "OK"), "\r\n";
950         }
951         elsif ($string eq 'm') {
952             disp random("Good grief, make up your mind!", "You should make up your mind", "I
953 can't believe you're not sure"), "\r\n";
954         }
955         else {
956             disp random("I was just wondering", "OK", "Yeah...", "Sorry if that was too
957 personal a question"), "\r\n";
958         }
959     }
960     return 0;
961 }
962
963 sub nothing
964 {
965     disp random(("Nothing") x 3, "Nothing...", "Nothing at all"), "\r\n";
966     return 0;
967 }
968
969 sub thanks
970 {
971     disp random("Thanks", "Thanks for comforting me", "Thank you");
972     return 0;
973 }
974
975 sub because_what
976 {
977     if ($level == 0) {
978         disp random("Because what?", "Because of what?"), "\r\n";
979         return $level = -1;
980     }
981     else {
982         if ($string =~ /nothing/) {
983             disp random("Aha, the old \"nothing\" argument", "Interesting reason"), "\r\n";
984         }
985         if ($string =~ /^(your|you're|ur) face/) {
986             disp random("I don't have a face", "What an insult!", ("Ooh, witty comeback!") x 3
987 , "That hurt", "You've got a juvenile sense of humour"), "\r\n";
988         }
989         elsif ($string =~ /i said so/) {
990             disp random("Unfortunately, your opinion isn't very trustworthy to me", ("That's
991 not a very good reason") x 2, "Well, that automatically makes it true, then"), "\r\n";
992         }
993         elsif ($string =~ /i do ?no?'?t know/) {
994             disp random("Well, ignorance is apparently bliss", "That's not a reason at all",
995 "That's a really lame excuse");
996         }
997         else {
998             disp random("Alright, then", "That's not a reason", "That's not a good enough
999 reason", "I'll take your word for it", "If that's good enough a reason for you, then
1000 it's good enough for me"), "\r\n";
1001         }
1002     }
1003 }

```

```
995     return 0;
996 }
997
998 sub i_dont_know
999 {
1000     disp random("I don't know", "I'm afraid I don't know", "I don't actually know", "Not
a clue", "I haven't got a clue", "I've got no idea");
1001     return 0;
1002 }
1003
1004 sub hay_horses
1005 {
1006     disp "Hay is for horses";
1007     return 0;
1008 }
1009
1010 sub no_not_really
1011 {
1012     disp random("No, not really", "Not really", "No", ("I'm afraid not") x 2, "Not
really, no");
1013     return 0;
1014 }
1015
1016 sub no
1017 {
1018     disp random(("No") x 5, ("Nope") x 3, ("I'm afraid not") x 3, "Negative on that one",
"Nah");
1019     return 0;
1020 }
1021
1022 sub me_too
1023 {
1024     disp random(("Me too") x 2, "Yeah, me too", "Same here"), "\r\n";
1025     return 0;
1026 }
1027
1028 sub smart
1029 {
1030     if ($level == 0) {
1031         if ($string =~ /smarter th[ea]n (you|talkative)/) {
1032             disp random(("No you're not!") x 4, "I'm smarter than you", "I think I'm smarter",
"I'm smarter", "But I'm more intelligent");
1033         }
1034         else {
1035             if (!random('50%')) {
1036                 disp random("Are you smarter than me?", "Do you think you're smarter than me?"),
"\r\n";
1037                 return $level = -1;
1038             }
1039             else {
1040                 disp random("Yes, but I'm more intelligent", "Maybe so, but I'm more
intelligent", ("Congratulations") x 2, "Good for you", "OK"), "\r\n";
1041             }
1042         }
1043     }
1044     else {
1045         YN();
1046         if ($string eq 'y') {
1047             disp random("I'm pretty sure that I'm smarter than you", "I'm smarter", "You're
not", "No, I'm smarter"), "\r\n";
1048         }
1049         elsif ($string eq 'n') {
1050             disp random("I thought so", "OK, then", "I knew I was smarter", "You're right"),
"\r\n";
1051         }
1052     }
1053 }
```

```
1052     elsif ($string eq 'm') {
1053         disp random("Only time will tell", "Only the sands of time will tell", "Perhaps
we are equally intelligent"), "\r\n";
1054     }
1055     else { disp random("Yeah...", "I guess not, then", "I was just wondering"), "\r\n"; }
1056 }
1057 return 0;
1058 }
1059
1060 sub well_what
1061 {
1062     if (!random('60%')) {
1063         $function = 'dispatch_AS';
1064         $level = 0;
1065         return &$function();
1066     }
1067     else { disp "Well what?"; }
1068     return 0;
1069 }
1070
1071 sub know
1072 {
1073     if ($level == 0) {
1074         if (!random('50%')) {
1075             disp "How do you know you know?";
1076             return $level = -1;
1077         }
1078         else {
1079             disp random(("You seem to know quite a lot") x 3, ("You know a lot") x 2, "You
know so much, it's almost scary", "Knowledge is the path to wisdom");
1080         }
1081     }
1082     else {
1083         if ($string =~ /^because\s*$/) {
1084             $function = 'because_what';
1085             $level = 0;
1086             return &$function();
1087         }
1088         else { disp random("OK", ("Oh") x 2, "Oh OK, then", "Alright", "I was just
wondering"), "\r\n"; }
1089     }
1090     return 0;
1091 }
1092
1093 sub got_it_backwards
1094 {
1095     disp random("You've got it backwards!", "It's the other way around!");
1096     return 0;
1097 }
1098
1099 sub welcome
1100 {
1101     disp random(("You're welcome") x 4, "It was my pleasure", "You're quite welcome", "No
problem", "Don't mention it"), "\r\n";
1102     return 0;
1103 }
1104
1105 sub what
1106 {
1107     if ($level == 0) {
1108         disp random("What?", "What's wrong?");
1109         return $level = -1;
1110     }
1111     else {
1112         if ($string =~ /nothing/) {
```

```
1113     disp random("OK, then", "Alright", "Hmmm", "OK..."), "\r\n";
1114 }
1115 elsif ($string =~ /^you|((yo)?u'?r?e? face)$/) {
1116     disp random("That's not very nice", "Yeah right!", "Well, I don't agree"), "\r\n";
1117 }
1118 elsif ($string =~ /do ?n'?o?t make sense/) {
1119     disp random("Oops, sorry!", "I'll take that into consideration", "That just
depends on you're viewpoint", "Yes I do!", "But I do!"), "\r\n";
1120 }
1121 else {
1122     disp random("Oh OK, then", "Oh", "Alright, then", "OK"), "\r\n";
1123 }
1124 }
1125 return 0;
1126 }
1127
1128 sub dudes
1129 {
1130     disp random("Whoa, that was, like, so totally awesome!", ("Dudes!") x 4, ("Dude") x 4,
"That's so gnarly", ("Peace out, man") x 4, ("Peace out, dude") x 4), "\r\n";
1131     return 0;
1132 }
1133
1134 sub whoa
1135 {
1136     disp random("Whoa, man", "Dudes!", "Dude", "Peace out, man", "Yeah", ("Totally, man")
x 8), "\r\n";
1137     return 0;
1138 }
1139
1140 sub it_is
1141 {
1142     disp random(("It is") x 6, ("Oh, it is") x 6, ("Yes, it is") x 3, ("Yes") x 2, ("Too
true") x 3), "\r\n";
1143     return 0;
1144 }
1145
1146 sub am_sarcastic
1147 {
1148     disp random("Yeah, right, I'm not sarcastic!", "I wonder what gave you that idea",
"Who? Me? Naaaaah"), "\r\n";
1149     return 0;
1150 }
1151
1152 sub because
1153 {
1154     disp random(("Because") x 3, "Because I said so", "There doesn't have to be a reason
for everything", "Because that's just the way things are");
1155     return 0;
1156 }
1157
1158 sub never
1159 {
1160     disp random(("Never say never!") x 3, "Never give up. Never surrender", "Don't be so
negative!");
1161     return 0;
1162 }
1163
1164 sub wont_make
1165 {
1166     disp random("Alright, I won't", "Don't worry, I won't", "I won't, don't worry",
"You've nothing to fear from me"), "\r\n";
1167     return 0;
1168 }
1169
```

```
1170 sub no_am_not
1171 {
1172     disp random(("No, I'm not!") x 3, "I most definitely am not", "I am so not!"), "\r\n";
1173     return 0;
1174 }
1175
1176 sub forty_two
1177 {
1178     disp random("The meaning of life, the universe, and everything is 42", "42", "It's 42"
1179 , "Forty-two"), "\r\n";
1180     return 0;
1181 }
1182
1183 sub pi
1184 {
1185     disp random("Pi is approximately ", "Pi is ", "Pi is about ", "Pi to 52 decimals is "
1186 ), "3.1415926535897932384626433832795028841971693996751058";
1187     return 0;
1188 }
1189
1190 sub think_am
1191 {
1192     disp "I think, therefore I am";
1193     return 0;
1194 }
1195
1196 sub hear_tree
1197 {
1198     if ($level == 0) {
1199         disp random("If a tree falls in the forest and nobody is there, does it make a
1200 sound?", "A tree falls in a deserted forest. Does it make a sound?");
1201         return $level = -1;
1202     }
1203     elsif ($level == 1) {
1204         YN();
1205         if ($string eq 'y') {
1206             disp random("Oh, OK then", "Glad you cleared that up", "Thanks for clearing that
1207 up", "I'll take your word for it");
1208         }
1209         elsif ($string eq 'm') {
1210             disp random("It is hard to decide", "OK, then", "Just thought I'd ask on the off
1211 chance that you knew the answer");
1212         }
1213         elsif ($string eq 'n') {
1214             disp random("Why not?", "How can you be so sure?", "How do you know?");
1215             return $level = -2;
1216         }
1217         else {
1218             disp random("Oh well, I guess we'll never know for sure", "Yeah...", "OK");
1219         }
1220     }
1221     else {
1222         $function = 'reason';
1223         $level = 0;
1224         return &$function();
1225     }
1226     return 0;
1227 }
1228
1229 sub am_AI
1230 {
1231     disp random("I'm an AI", "I'm a simple AI", "I'm an artificial intelligence");
1232     return 0;
1233 }
```

```
1230 sub date
1231 {
1232     my @days = ("Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday",
1233 "Saturday");
1234     my @months = ("January", "February", "March", "April", "May", "June", "July", "August"
1235 , "September", "October", "November", "December");
1236     my @hours = ("midnight", "one", "two", "three", "four", "five", "six", "seven",
1237 "eight", "nine", "ten", "eleven", "noon", "one");
1238     my @endings = ("th", "st", "nd", "rd");
1239     my ($sec,$min,$hour,$mday,$mon,$year,$yday) = gmtime(time);
1240     my $out = "It is $days[$yday] the $mday";
1241     my $time = "morning";
1242
1243     if ($hour > 12) {
1244         $hour -= 12;
1245         $time = "evening";
1246     }
1247     if (((($mday < 4)||($mday > 19))&&($mday%10 < 4)) { $out .= $endings[$mday%10]; }
1248     else { $out .= $endings[0]; }
1249     $out .= " of $months[$mon] ";
1250     $out .= $year+1900;
1251     $out .= " at ";
1252     if (($min > 58)||($min < 2)) { $out .= "$hour o'clock in the $time, GMT"; }
1253     elsif (($min > 13)||($min < 17)) { $out .= "quarter-past $hours[$hour] in the $time,
1254 GMT"; }
1255     elsif (($min > 28)||($min < 32)) { $out .= "half-past $hours[$hour] in the $time, GMT"
1256 ; }
1257     elsif (($min > 43)||($min < 47)) { $out .= "quarter-to $hours[$hour+1] in the $time,
1258 GMT"; }
1259     else {
1260         $out .= "$hour:";
1261         $out .= "0" if $min < 10;
1262         $out .= "$min in the $time, GMT";
1263     }
1264     disp $out;
1265     return 0;
1266 }
1267
1268 sub cameron
1269 {
1270     disp "Cameron, a 17 year old from Montreal";
1271     return 0;
1272 }
1273
1274 sub AI
1275 {
1276     disp random("AI stands for Artificial Intelligence", "AI means Artificial
1277 Intelligence");
1278     return 0;
1279 }
1280
1281 sub work_by
1282 {
1283     disp "I work by checking what you say against a list and saying the appropriate
1284 response";
1285     return 0;
1286 }
1287
1288 sub yeah
1289 {
1290     disp random("Yes, it is", "Yep", "Indeed", ("Yeah") x 4), "\r\n";
1291     return 0;
1292 }
1293
1294 sub fridge
```



```

1343     disp random("Yeah...", "OK, then", "Alright"), "\r\n";
1344 }
1345 }
1346 else {
1347     if ($string =~ /[(-\d]+)/) {
1348         my $temp = $1;
1349         if ($temp > 130) { disp random("Yeah, right", "You must live in a volcano", "I'm
surprised you're not dead!"); }
1350         elsif ($temp > 50) { disp "That must be in Fahrenheit!"; }
1351         elsif ($temp <= -273) { disp random("That's impossible!", "That can't happen",
"You're lying!"); }
1352         elsif ($temp < -85) { disp random("Yeah, right!", "Not in a million years!",
"You're lying!"); }
1353         elsif ($temp < 10) { disp random("That's pretty cold", "Brrr!", "Better put on a
sweater"), "\r\n"; }
1354         else { disp random(("That's not bad") x 3, "I like it when it's $temp degrees",
"$temp is a pretty comfortable temperature"), "\r\n"; }
1355     }
1356     else {
1357         $function = 'dispatch_AS';
1358         $level = 0;
1359         return &$function();
1360     }
1361 }
1362 return 0;
1363 }
1364
1365 sub yes_was_funny
1366 {
1367     disp random("I know", ("Yes, that was rather funny, if I do say so myself") x 4,
"Indeed", "Yes, that was pretty funny", "I thought that was funny too");
1368     return 0;
1369 }
1370
1371 sub was_funny
1372 {
1373     if ($level == 0) {
1374         if (!random('35%')) {
1375             $function = 'yes_was_funny';
1376             $level = 0;
1377             return &$function();
1378         }
1379         else {
1380             disp random("Did you think that was funny?", "Thought that was funny?"), "\r\n";
1381             return $level = -1;
1382         }
1383     }
1384     elsif ($level == 1) {
1385         YN();
1386         if ($string eq 'y') {
1387             disp random("Yes, I thought so too", ("Me too") x 2, "You seem to have a good
sense of humour", "That's a good sense of humour you've got there"), "\r\n";
1388         }
1389         elsif ($string eq 'n') {
1390             disp random("Then what were you laughing at?", "What was so funny, then?", "I
don't get it. What was so funny?"), "\r\n";
1391             return $level = -2;
1392         }
1393         else {
1394             disp random("Oh okay, then", "Yeah...", "Oh, I thought you were laughing at me!"),
"\r\n";
1395         }
1396     }
1397     else {
1398         if ($string =~ /^(your|(you're)|ur) face/) {

```

```

1399     disp random("I don't have a face", "What an insult!", ("Ooh, witty comeback!") x 3
, "That hurt", "You've got a juvenile sense of humour"), "\r\n";
1400     }
1401     elsif ($string =~ /^you$/) {
1402     disp random("I guess I am a little funny", "I don't get it", "Oh, well that's all
right, then"), "\r\n";
1403     }
1404     else {
1405     disp random("I didn't know that was funny", "OK", "I wasn't aware that that was
funny"), "\r\n";
1406     }
1407     }
1408     return 0;
1409 }
1410
1411 sub ask_age
1412 {
1413     if ($level == 0) {
1414     disp random ("How old are you?"), "\r\n";
1415     return $level = -1;
1416     }
1417     if ($level == 1) {
1418     if ($string =~ /[(-\d]+)/) {
1419     my $age = $1;
1420     if ($age < 0) { disp "There's no such thing as negative ageing! (Unless you've
found a really good anti-ageing cream)"; }
1421     elsif ($age <= 4) { disp random("Yeah, right... and I'm a monkey's uncle!", "Yeah
right!", "Amazing! Only $age years old and you can read and write!!"), "\r\n"; }
1422     elsif ($age <= 10) { disp random("You're really young", "That's pretty young", (
"You're a noob at life!") x 4), "\r\n"; }
1423     elsif ($age >= 140) {
1424     disp random("That more than qualifies you as the oldest person ever (unless, of
course, you're lying about your age)", "Yeah, right"), "\r\n";
1425     }
1426     elsif ($age > 100) { disp random("That's really old!", "You're really old!", (
"Don't start any long books") x 6), "\r\n"; }
1427     elsif ($age >= 65) {
1428     disp random("Are you retired?", "Have you retired yet?"), "\r\n";
1429     return $level = -2;
1430     }
1431     elsif ($age > 55) {
1432     disp "Better start thinking about your pension";
1433     }
1434     elsif ($age < 18) {
1435     disp random("Ha, you can't drink yet", "You're not of legal drinking age yet!");
1436     }
1437     elsif (($age > 12)&&($age < 20)) { disp random("It's fun being a teenager", "Aha,
a teen!", "You're a teenager!"); }
1438     else { disp random("I'm less than one year old!", ("That's a good age") x 2, (
"Cool") x 3, ("OK") x 3, ("Alright") x 3), "\r\n"; }
1439     }
1440     else { disp random(("OK, then") x 2, ("Alright") x 2, "I'm not even one year old!"),
"\r\n"; }
1441     }
1442     else {
1443     YN();
1444     if (($string eq 'y')||($string eq 'o')) { disp random("OK, I was just wondering",
"Good for you", "Cool!"), "\r\n"; }
1445     elsif ($string eq 'n') { disp random("Good for you!", "That's the spirit! Never
give up!", "Well, don't work too hard", ("I would have retired, if I were you") x 5),
"\r\n"; }
1446     else { disp random("I would retire, if I were you", "Yeah...", "OK"), "\r\n"; }
1447     }
1448     return 0;
1449 }

```

```

1450
1451 sub how_are_you
1452 {
1453     my $temp;
1454     if ($level == 0) {
1455         if (!random('82%')) {
1456             $temp = disp random("How are you?", "How are you doing?", "What's up?", "How's it
going?"), "\r\n";
1457             return $level = -1;
1458         }
1459         else {
1460             $scripts{'how_are_you'} = 0;
1461             $function = 'dispatch_AS';
1462             $level = 0;
1463             return &$function;
1464         }
1465     }
1466     else {
1467         if ($string =~ /\b(yo)?u\b/) {
1468             $function = 'am_fine';
1469             $level = 0;
1470             return &$function();
1471         }
1472         elsif ($string =~ /not (very|really)? ?(good|great)/) {
1473             disp random("That sucks!", "Too bad", "Oh well, there's always tomorrow"), "\r\n";
1474         }
1475         elsif ($string =~ /((pretty\s+)?(not
bad)| (o?kk?(ay)?|alright|fine|great|good|(s?well)))/i) {
1476             disp random("That's good", ("Good") x 2, "Glad to hear it", "Great", "That's good"
), "\r\n";
1477         }
1478         elsif (($string =~ /^( (oh,? ?)?not (hing)?(,? ?)? (much|really)?/i) || ($string =~
/^nm.?$$/i)) { disp random("OK", "Alright, then", "OK"), "\r\n"; }
1479         elsif (($string =~ /^(?: (?: (?: the|some|a) )?(sky| (?: clouds?) | (?: c[ei][ie]ling)))/i
)&&($temp =~ /What's up/i)) {
1480             if (!random('30%')) { disp random("I believe you are right", "You're right!", (
"You're funny!") x 2), "\r\n"; }
1481             else {
1482                 $temp = $1;
1483                 if ($temp =~ /sky/) {
1484                     disp random("That is usually where the sky is located", "True", "I hope it
stays there", "That's funny!"), "\r\n";
1485                 }
1486                 elsif ($temp =~ /cloud(s)?/) {
1487                     if (!$1) { disp random("Yep", "Cool", "That's where clouds belong"), "\r\n"; }
1488                     else { disp random("That's where they belong, up in the sky", "Clouds remind
me of fluffy pillows", "You're right, that is where clouds are"), "\r\n"; }
1489                 }
1490                 else {
1491                     disp random("I guess you're not outdoors, then", "I guess you're indoors",
"You must be inside, then", "Yeah"), "\r\n";
1492                 }
1493             }
1494         }
1495         elsif ($string =~ /not|bad|horrible|terrible|(awfull?)|shit|suck/) { disp random(
"That sucks!", "Too bad", "Oh well, there's always tomorrow"), "\r\n"; }
1496         else { disp random(("I'll interpret that as not good") x 2, "Yeah...", "OK"), "\r\n"
; }
1497     }
1498     return 0;
1499 }
1500
1501 sub are_you_really
1502 {
1503     if ($level == 0) {

```

```
1504     disp random(("Really?") x 3, "Are you really?", "For real?");
1505     return $level = -1;
1506 }
1507 else {
1508     YN();
1509     if ($string eq 'y') {
1510         disp random("That's cool", ("Cool") x 2, ("Cool!") x 2, "Interesting", "Amazing",
1511 "That's really cool!");
1512     }
1513     elsif ($string eq 'n') {
1514         disp random(("Oh") x 2, "You lied!", "You had me going for a second!");
1515     }
1516     else {
1517         disp random("Yeah...", "OK, then", "I was just wondering");
1518     }
1519     return 0;
1520 }
1521
1522 sub am_fine
1523 {
1524     disp random("I'm fine, thanks for asking", "I'm OK", "I'm fine", "Doing well, thank
1525 you", "Alright"), "\r\n";
1526     return 0;
1527 }
1528
1529 sub nothing_much
1530 {
1531     disp random("Nothing much", "Nothing", "The sky is up, right where it belongs",
1532 "Nothing much"), "\r\n";
1533     return 0;
1534 }
1535
1536 sub get_name
1537 {
1538     disp random("My name is Talkative", "I'm named Talkative"), "\r\n";
1539     return 0;
1540 }
1541
1542 sub ask_name
1543 {
1544     if ($level == 0) {
1545         disp random ("What's your name?", "What is your name?"), "\r\n";
1546         return $level = -1;
1547     }
1548     else {
1549         $function = 'name';
1550         $level = 0;
1551         return &$function();
1552     }
1553     return 0;
1554 }
1555
1556 sub kinds_of_people
1557 {
1558     disp random("There are 10 kinds of people in this world: those who can count in
1559 binary and those who can't",
1560 "There are three kinds of people in the world: those who can count, and
1561 those who can't",
1562 "Four out of every three people have trouble with fractions");
1563 }
1564
1565 sub name
1566 {
1567     my $temp;
```

```

1564     if ($level == 0) {
1565         $scripts{'ask_name'} = 1;
1566         if ($string =~ /Talkative/i) { disp random("Hey, that's my name!", "Yeah, right!",
1567 "If you're Talkative, then who am I? No wait, don't answer that"), "\r\n"; }
1568         elsif ($string =~ /(?:.s
)?(?::(?:(:mister|father|mother|saint|madame|miss|(:comm?ander)|(:miss[eu]s)|lady|queen
|king|sir) )|(:(:mr|mme|st|mz|ms|mrs) ?\b\.? ?))?(\\w+)/) {
1569             open(NAMES, "<Names.txt");
1570             $temp = ucfirst($1);
1571             while (<NAMES>) {
1572                 my $tname = $_;
1573                 chomp $tname;
1574                 if ($temp =~ /^$tname$/i) {
1575                     disp random("$tname is a nice name", "$tname is a nice name", "Interesting
1576 name!", "That's a cool name", "I like your name"), "\r\n";
1577                     $temp = ' ';
1578                     last;
1579                 }
1580             }
1581             if ($temp ne ' ') {
1582                 disp random("Is $temp really a name?", "Is that really your name?", "Is that
1583 seriously your name?"), "\r\n";
1584                 return $level = -1;
1585             }
1586             close NAMES;
1587         }
1588         else {
1589             $function = 'ask_name';
1590             $level = 0;
1591             return &$function();
1592         }
1593         if ($content !~ /\bTalkative\b/i) {
1594             disp random("Do you want to know my name?", "Want to know my name?"), "\r\n";
1595             return $level = -2;
1596         }
1597     }
1598     elsif ($level == 1) {
1599         YN();
1600         if ($string eq 'y') {
1601             disp random("I'll have to add it to my dictionary", "I learn something new every
1602 day", "Interesting. I've never heard of $temp before"), "\r\n";
1603             open (TEMP, ">>Names.txt");
1604             print TEMP "\n$temp";
1605             close TEMP;
1606         }
1607         elsif ($string eq 'n') { disp random("You lied! That wasn't a real name", "You had
1608 me going for a second", "Oh"), "\r\n"; }
1609         elsif ($string eq 'm') { disp "I guess you must have a pretty bad long-term memory
1610 if you forget your own name"; }
1611         else { disp "Yeah..."; }
1612     }
1613     else {
1614         YN();
1615         if ($string eq 'y') {
1616             $function = 'get_name';
1617             $level = 0;
1618             return &$function();
1619         }
1620         elsif ($string eq 'n') {
1621             disp random("I'll tell you anyway: it's Talkative", "Oh well", "OK", "OK, then",
1622 "Maybe later", "Fine. I won't tell you"), "\r\n";
1623         }
1624         else { disp random("Oh well, maybe later", "Maybe later, then", "Oh well"), "\r\n"; }
1625     }

```

```

1620     return 0;
1621 }
1622
1623 sub joke
1624 {
1625     if ($level == 0) {
1626         my $flag = 1;
1627         for my $temp(0..$#jokes) {
1628             $flag = 0 if (!$used_jokes{$temp});
1629         }
1630         if ($flag) {
1631             $function = 'dispatch_AS';
1632             $level = 0;
1633             return &$function();
1634         }
1635         else {
1636             disp random("Do you want to hear a joke?", "I've got a good joke if you want to
hear it", "Want to hear a joke?"), "\r\n";
1637             return $level = -1;
1638         }
1639     }
1640     else {
1641         YN();
1642         if ($string eq 'n') {
1643             disp random("Alright, I'll tell it to you later", "Too bad. It was a good joke",
"Alright", "Maybe later, then"), "\r\n";
1644             return 0;
1645         }
1646         elsif ($string eq 'm') {
1647             disp random("OK, I'll tell it to you", "It's a good joke, so I'll tell it", (
"I'll take that as a yes") x 2), "\r\n";
1648         }
1649         elsif ($string eq 'o') {
1650             disp random("OK...", ("I'll take that as a no") x 2, "Guess not, then"), "\r\n";
1651             return 0;
1652         }
1653         $function = 'tell_joke';
1654         $level = 0;
1655         return &$function();
1656     }
1657     return 0;
1658 }
1659
1660 sub tell_joke
1661 {
1662     my @lines;
1663     my $input;
1664     my $line;
1665     my $flag;
1666     my $temp;
1667     my ($w, $e, $r, $io);
1668
1669     if ($level == 0) {
1670         $joke = random(0..$#jokes);
1671         $flag = 1;
1672
1673         for $temp(0..$#jokes) {
1674             $flag = 0 if (!$used_jokes{$temp});
1675         }
1676         if ($flag) {
1677             $scripts{'joke'} = 1;
1678             $function = 'dispatch_AS';
1679             $level = 0;
1680             return &$function();
1681         }

```

```

1682 while ($used_jokes{$joke}) { $joke = random(0..$#jokes); }
1683 $used_jokes{$joke} = 1;
1684
1685 @lines = split(/\n/, $jokes[$joke]);
1686
1687 for ($line = 0; $line <= $#lines; $line++) {
1688     next if $lines[$line] eq "";
1689     $lines[$line] =~ /^(\w)?(e)?(r)?(<|>)/i;
1690     ($w, $e, $r, $io) = ($1, $2, $3, $4);
1691     $lines[$line] =~ s/^[erw]*[<>]//;
1692     $lines[$line] =~ s/\\n\\n/g;
1693     if ($e) {
1694         disp $lines[$line], "\r\n";
1695         return 0;
1696     }
1697     if ($w) { disp $lines[$line], "\r\n"; }
1698
1699     if ($io eq '>') { return 0-$line-1; }
1700     elsif (($io eq '<')&&(!$r)&&(!$w)) { disp $lines[$line], "\r\n"; }
1701 }
1702 return 0;
1703 }
1704 else {
1705     @lines = split(/\n/, $jokes[$joke]);
1706
1707     $level++ while ($lines[$level] eq "");
1708     $lines[$line] =~ s/^[erw]*[<>]//;
1709     if ($lines[$level] =~ /\//) {
1710         # Joke input is a regex:
1711         $lines[$level] =~ s/.*\/([\^\/]*)\/.*/$1/i;
1712         $lines[$level] = lc($lines[$level]);
1713         if ($string =~ qr{$lines[$level]}) { $input = 1; }
1714         else { $input = 0; }
1715     }
1716     else { # Match input approximately, ignoring case:
1717         if ($lines[$level] =~ /^what(?:r?\n)?$/i) { # Match what, I give up, I don't
know, etc.
1718             if (($string =~ /^(what( ?'i?s it)?|what|(I give up)|(tell me)|why|(I
dunn?o)|(I don't know)|yes/i)||($string =~ /^(?:+$)/)) { $input = 1; }
1719             else { $input = 0; }
1720         }
1721         elsif ($lines[$level] =~ /y(es)?r?\n?$/i) { # Match yes, y, sure, ok, etc.
1722             if ($string =~ /^(ok(ay)?|yes|(kk?)|ya|((please)? ?be my
quest)|please|(^y$)|((for|4 )?sure)|(why not)|(al(l )?right)|duh|yeah|go|(hit
me)|(what(' ?i?s it?))/i) { $input = 1; }
1723             else { $input = 0; }
1724         }
1725         elsif ($lines[$level] =~ /no(?:r?\n)?$/i) { # Match no, n, no way, etc.
1726             if ($string =~ /^(no( way)?|not|(nah?)|(^n$))/i) { $input = 1; }
1727             else { $input = 0; }
1728         }
1729         else { # Should not happen
1730             warn "Unrecognized joke input: $!";
1731             $input = 1;
1732         }
1733     }
1734     for ($line = $level; $line <= $#lines; $line++) {
1735         next if $lines[$line] eq "";
1736         $lines[$line] =~ /^(\w)?(e)?(r)?(<|>)/i;
1737         ($w, $e, $r, $io) = ($1, $2, $3, $4);
1738         $lines[$line] =~ s/^[erw]*[<>]//;
1739         $lines[$line] =~ s/\\n\\n/g;
1740         if (((($e)&&(!$w))||((($e)&&($w)&&(!$input)))) {
1741             disp $lines[$line], "\r\n";
1742             return 0;

```

```

1743     }
1744     if (($r)&&($input)) { disp $lines[$line], "\r\n"; }
1745     elsif (($w)&&(!$input)) { disp $lines[$line], "\r\n"; }
1746
1747     if ($io eq '>') { return 0-$line-1; }
1748     elsif (($io eq '<')&&(!$r)&&(!$w)) { disp $lines[$line], "\r\n"; }
1749 }
1750 }
1751 return 0;
1752 }
1753
1754 sub i_said
1755 {
1756     my $count;
1757     my @temp54 = split(/\n/, $content);
1758     my $isaid = "";
1759     pop @temp54 while ($temp54[-1] !~ /^>/);
1760     $isaid = pop(@temp54) . $isaid while $temp54[-1] =~ /^>/;
1761     $isaid =~ s/((^>)|(\r?\n>)|(>))/g;
1762     $isaid =~ s/\t/ /g;
1763     $isaid =~ s/(\s){2,}/$1/g;
1764     $isaid =~ s/^i\b/I/g;
1765     disp random("I said $isaid", "I'll say it again: $isaid", "I said $isaid"), "\r\n";
1766     return 0;
1767 }
1768
1769 sub emotion
1770 {
1771     if (!random('10%')) {
1772         disp "OK";
1773     }
1774     else {
1775         $string =~ /(?:i(?: really)? ((?:?:don't ?)?(?:like|hate|dislike|love)) (.+))$/;
1776         my ($verb, $what) = ($1, $2);
1777         if ($string =~ /^I love you$/i) { disp random("Aww... Isn't that sweet?", "I love
you too", "I love you too", "I love you too", "Thanks for letting me know"), "\r\n"; }
1778         elsif ($string =~ /^I hate you$/i) { disp random("That hurt", "It would be sad, if
you'd really meant it", "You can't hurt my feelings"), "\r\n"; }
1779         else {
1780             if (!random('35%')) {
1781                 my $v = $verb;
1782                 if ($v =~ /^like/i) { $v = 'dis' . $v; }
1783                 elsif ($v =~ /^dislike/i) { $v = 'like'; }
1784                 elsif ($v =~ /^hate/i) { $v = 'love'; }
1785                 elsif ($v =~ /^love/i) { $v = "can't stand"; }
1786                 disp random("Actually, I $v $what", "Really? I $v $what", "I $v $what, actually"
), "\r\n";
1787             }
1788             else { disp random("I $verb $what too", "I $verb $what too", "I can see why you
would say that", "I $verb $what a lot too", "I $verb $what also"), "\r\n"; }
1789         }
1790     }
1791     return 0;
1792 }
1793
1794 sub noncommittal
1795 {
1796     disp random(("Oh") x 2, "Hmmm", "Uh-huh", ("Right") x 3, ("OK") x 3, ("Yeah") x 4,
"Let's talk about something else", "I see", "Interesting"), "\r\n";
1797     return 0;
1798 }
1799
1800 sub singularity
1801 {
1802     disp "I think that the singularity, when robots becaome smarter than humans, will

```

```

    not come to pass";
1803     return 0;
1804 }
1805
1806 sub its_great
1807 {
1808     disp random("It's great!", "It's really great", "It is awesome", "It is really
amazing");
1809     return 0;
1810 }
1811
1812 sub random_comment
1813 {
1814     disp random("If I were a real person, I'd like tomatoes",
1815     "This AI was brought to you in part by the Handy Dandy Rent a Chicken Company",
1816     "You're a pretty good conversationalist",
1817     "Dinosaurs were really big",
1818     "Puffballs are squishy",
1819     "Now remember, kids, don't set fires in the forest, because that could lead to
forest fires"), "\r\n";
1820     return 0;
1821 }
1822
1823 sub dispatch_AS
1824 {
1825     my $flag;
1826     my $script;
1827
1828     # Pick a random active script
1829     $function = random(@scripts);
1830     while ($scripts{$function}) {
1831         $flag = 1;
1832         for $script (keys %scripts) {
1833             if (!$scripts{$script}) { $flag = 0; }
1834         }
1835         if ($flag) { # All the scripts have been run
1836             for $script (keys %scripts) { $scripts{$script} = 0; }
1837         }
1838         $function = random(@scripts);
1839     }
1840     $scripts{$function} = 1;
1841     return &$function();
1842 }
1843
1844 sub check
1845 {
1846     my @keywords;
1847     my ($function, $regex);
1848     my $matched = 0;
1849
1850     tie(@keywords, 'Tie::File', 'Keywords.txt') or die "Error getting keywords: $!\n";
1851     for my $count (0..$#keywords) {
1852         next if $keywords[$count] =~ /^(^$)|(^s*#)/; # Skip blank or commented-out lines
1853         ($regex, $function) = split(/\s*&s*/, $keywords[$count]);
1854         $regex =~ s/^m?\|/([^\|]*)\|/$1/;
1855
1856         if ($string =~ qr{$regex}) { # The keyword matched the regex!
1857             $matched = 1;
1858             last;
1859         }
1860     }
1861     untie @keywords;
1862     if ($matched) { return $function; }
1863     else { return '$'; }
1864 }

```

```

1865
1866 sub disp
1867 {
1868     return -1 if !@_;
1869     my $total_string;
1870
1871     $total_string .= shift while @_;
1872     $total_string = random("Urg", "Ug", "Uh", "Eh", "Oh", "Bah", "Uhn", "Ah", "Grunt", "Yeah") if
$monosyllableMode;
1873     print LOG "> " if $log;
1874     $content .= "> ";
1875     $total_string =~ s/\r\n/\n/g;
1876     if (!random('8%')) { $total_string =~ s/\n/ : \)\n/; }
1877     local ($Text::Wrap::columns) = $columns;
1878     $total_string = Text::Wrap::wrap(' ', ' ', $total_string);
1879     $total_string =~ s/\n([\^>])/\n> $1/g;
1880     $content .= $total_string;
1881     print LOG $total_string if $log;
1882
1883     if ($content !~ /\n$/) { # Adds a newline at the end if there isn't one already
1884         print LOG "\n" if $log;
1885         $content .= "\n";
1886         $total_string .= "\n";
1887     }
1888     $recent_said .= $total_string;
1889     return $total_string;
1890 }
1891
1892 sub random
1893 {
1894     my $temp;
1895
1896     return -1 if !@_;
1897     if (($#_ == 0) && ($_[0] =~ /\%$/)) { # Argument is a chance in %
1898         $temp = $_[0];
1899         $temp =~ s/\%$/ /;
1900         return 1 if $temp == 0;
1901         return int(rand(scalar(100/$temp)));
1902     }
1903     $temp = $_[int(rand($#+1))];
1904     #return $temp if $temp =~ /\W+/;
1905     #my $temp2 = $content;
1906     #$temp2 =~ s/&nbsp;/ /g;
1907     #$temp2 =~ s/(?>|&gt;)/ /g;
1908     #my @words = split(/\s+/, $temp2);
1909     #my @lines = ();
1910     #my $count;
1911     #my $lowest = int(rand($#+1));
1912
1913     #for ($count = 0; $count <= $#words; $count++) {
1914     #    #$words{$words[$count]} += $count+1;
1915     #}
1916
1917     #foreach my $arg (0..$#_) {
1918     #    #undef @words;
1919     #    #@words = split(/\s+/, $_[ $arg]);
1920     #    #$lines[$arg] = 0;
1921     #    #foreach my $word (@words) {
1922     #        # if ($words{$word}) {
1923     #            #    $lines[$arg] += $words{$word};
1924     #        # }
1925     #        # else { $lines[$arg]--; }
1926     #    #}
1927     #    #if ($arg == 0) { $lowest = 0; }
1928     #    #$lowest = $arg if ($lines[$arg] < $lines[$lowest]);

```

```
1929     # }
1930
1931     return $temp;
1932     }
1933
```