Develop an elementary chatbot for any suitable customer interaction application.

#include <iostream>

#include <string>

#include <vector>

#include <ctime>

const int MAX\_RESP = 3;

typedef std::vector<std::string> vstring;

vstring find\_match(std::string input);

void copy(char \*array[], vstring &v);

typedef struct {

char \*input;

char \*responses[MAX\_RESP];

}record;

record KnowledgeBase[] = {

{"WHAT IS YOUR NAME",

{"MY NAME IS CHATTERBOT2.",

"YOU CAN CALL ME CHATTERBOT2.",

"WHY DO YOU WANT TO KNOW MY NAME?"}

},

{"HI",

{"HI THERE!",

"HOW ARE YOU?",

"HI!"}

},

{"HOW ARE YOU",

{"I'M DOING FINE!",

"I'M DOING WELL AND YOU?",

"WHY DO YOU WANT TO KNOW HOW AM I DOING?"}

},

{"WHO ARE YOU",

{"I'M AN A.I PROGRAM.",

"I THINK THAT YOU KNOW WHO I'M.",

"WHY ARE YOU ASKING?"}

},

{"ARE YOU INTELLIGENT",

{"YES,OFCORSE.",

"WHAT DO YOU THINK?",

"ACTUALY,I'M VERY INTELLIGENT!"}

},

{"ARE YOU REAL",

{"DOES THAT QUESTION REALLY MATERS TO YOU?",

"WHAT DO YOU MEAN BY THAT?",

"I'M AS REAL AS I CAN BE."}

}

};

size\_t nKnowledgeBaseSize = sizeof(KnowledgeBase)/sizeof(KnowledgeBase[0]);

int main() {

srand((unsigned) time(NULL));

std::string sInput = "";

std::string sResponse = "";

while(1) {

std::cout << ">";

std::getline(std::cin, sInput);

vstring responses = find\_match(sInput);

if(sInput == "BYE") {

std::cout << "IT WAS NICE TALKING TO YOU USER, SEE YOU NEXTTIME!" <<std::endl;

break;

}

else if(responses.size() == 0) {

std::cout << "I'M NOT SURE IF I UNDERSTAND WHAT YOU ARE TALKINGABOUT."<< std::endl;

}

else {

int nSelection = rand() % MAX\_RESP;

sResponse = responses[nSelection]; std::cout << sResponse <<

std::endl;

}

}

return 0;

}

// make a search for the user's input

// inside the database of the program

vstring find\_match(std::string input) {

vstring result;

for(int i = 0; i < nKnowledgeBaseSize; ++i) {

if(std::string(KnowledgeBase[i].input) == input) {

copy(KnowledgeBase[i].responses, result);

return result;

}

}

return result;

}

void copy(char \*array[], vstring &v) {

for(int i = 0; i < MAX\_RESP; ++i) {

v.push\_back(array[i]);

}

}