

FREQUENTLY ASKED QUESTIONS

/******
/*****

**This forum and the code it contains are meant to be used ONLY by members of the LaGuardia Computer Programming Club.
**Please contact Mr. Moore, the faculty advisor of the club, at englishmoore@englishmoore.com if you are using them in connection with any other group or class. Thank you.

FREQUENTLY ASKED QUESTIONS

Special note: Please post all of your source code on the board in this section so that we can work on it during club meetings. Enjoy!

1. How do I make a computer program?

You must create a file with code in a programming language that the computer understands. First, pick your language. Our club will be dealing primarily with C++, but you can also use C, Java, Basic, HTML, Scheme, etc. Second, pick your compiler. I'd recommend using the free one that Dev C++ provides. That way you will be sure that your code is compatible with that of the other club members. Third, begin to type your program.

2. How do I download and install the Dev C++ compiler?

(5-10 minutes)

Enter the following URL into your Web browser: http://prdownloads.sourceforge.net/dev-cpp/devcpp-4.9.9.2_setup.exe Click on Download. Run the setup file by double clicking on it, and follow the prompts. It takes about one minute to install. Run the program, and go through the first time configuration prompts. This will take another two minutes.

3. How do I start a new .cpp file, how do I compile it, and how do I run the executable that it produces?

File->New->Source File creates a new .cpp file. Click on Execute->Compile to compile. Click on Execute->Run to run your program. Note that your program will not run until it has first been compiled successfully.

4. What do I do when I get an error message?

Post your code in this section so that other club members can give you feedback.

5. What if I still need more help?

Your code must be 100% error free or it will not compile. Review the code for any small errors. Common errors include forgetting a bracket "}," parenthesis ")," or semicolon ";" Be sure that the functions that you are using can be found in the libraries you are including. Try commenting out code until you identify the problematic area (*/*This is commented out.**). Remember to compile, run, and save often.

6. Where can I look for more assistance?

Newsgroups are extremely helpful. Try going to groups.google.com. Type in the exact error message that you are getting or the name of a function, and you will probably get some useful advice.

7. What are some sample programs that I can test?

View the sample programs on this board.

8. How do I post on this message board?

See the "How Do I Use This Message Board" section.

JOIN THE LAG COMPUTER CLUB

*/**

//Dear Prospective Computer Club Member,

//J01N: Click on "Add Reply" above. Simply post your e-mail address to join the club.

//1NTR0DUCT10N: Welcome! Our club is dedicated to helping each member make video games! We have fifteen members and are growing fast. It is never too late to join the club. Be sure to tell your friends about us.

//M3T1NG T1M3: We meet every Thursday after school in the fourth floor computer lab. We will occasionally move next door for a few minutes so that we can discuss in quiet some of the complex issues that come up when programming in C++. if you don't find us in the lab, try one of the adjacent classrooms.

//F1R5T 5T3P: Read the F.A.Q. in the Help Center. This will explain how to download the compiler program to your home computer that will let you create games.

//WHAT'5 N3XT: Work at your own pace, and post questions in the "ask for help" section. Click on Part 1. Copy and paste this code into the Dev-C++ compiler. Compile and run it. Review and modify it until you feel like you understand how it works. Next, create your own original AI (artificial intelligence) program that produces the output at the bottom of the message. Finally, continue to Part 2.

//1NV1T1NG FR13ND5: Feel free to invite friends to the club. it is certainly not too late to get involved.

//P05T1NG C0D3: Thank you for continuing to post your latest source code in the Help Center (englishmoore.com – password: computer). This can help you (we can give you feedback), it can help other members (they can learn from your code), and it helps the club as a whole (busier board->more fun for everyone). You are also always welcome to e-mail code or questions to laguardiagamemakers@yahoogroups.com or englishmoore@englishmoore.com.

//THANK5, and I look forward to helping you make games soon! :-)

Best wishes,

Mr. Moore

*/

PART ONE: HELLO WORLD

/*****

**This forum and the code it contains are meant to be used ONLY by members of the LaGuardia Computer Programming Club.

**Please contact Mr. Moore, the faculty advisor of the club, at englishmoore@englishmoore.com if you are using them in connection with any other group or class. Thank you.

>>Skill: Running and compiling a program in C++

>>Task: Review the program below. Copy and paste it into the compiler. Compile and run it. View the message board if you have any problems with this. Next, create your own version of this program named yourName.cpp that introduces your AI.

>>Extra help: Review the information sheet at the bottom of this program.

>>Program 1: Hello World:

```
*/
```

```
#include <stdio.h>
```

```
#include <conio.h>
```

```
int main(){
```

```
printf("Hello World!");
```

```
getch();
```

```
return 0;
```

```
};
```

```
/*
```

>>Desired output of your new program:

```
*****
** .....**
** .....yourAI version 1.0.....**
** .....**
*****
```

>>For further research: Look into gets() and puts().

Information sheet – Program 1 – Hello World!

Computer Programming Club, LaGuardia Arts High School

Mr. Moore, englishmoore.com

Introduction: Welcome to the exciting world of computer programming! Today you will write your first program, and soon you will write your first game!

Skills: Compiling and running a C++ program

It is difficult to make and run computer programs. If you use the powerful C++ language, you will do the following:

- 1- Type and save your C++ file (yourFile.cpp).
- 2- Run a compiler that translates your code to machine code (a series of 1's and 0's that the computer can understand).

You can now double-click on your new program (yourFile.exe). You can upload, download, and e-mail this program to anyone in the world!

Functions: There are many different action words that you can use in your program to help you make the game you want. You can tell a word is a function if it has parentheses after it ().

Consider a function to be similar to a verb. A function is a predefined word that lets you perform an action in a program. The following are some predefined functions in C++:

printf() – prints words

scanf() – inputs words and numbers from the user

gets() – inputs words from the user

getch() – inputs one letter or number from the user

puts() – prints words

malloc()- sets aside memory inside of the computer for a purpose

dragonAttack() – a function that you can create to make a dragon in your game attack.

As you can see, you can also create your own functions!!

Arguments: Sometimes words and numbers go inside of the parentheses. These are called arguments. Some functions accept 0, 1, 2, or even more arguments. For example, gets(arg1) accepts one argument, and getch() accepts no arguments.

Libraries: Predefined functions are stored in libraries. Whenever you include a function in your source code, you also have to make sure that you type the following line at the very top of your program:

```
#include <libraryThatGoesWithMyFunction.h>
```

For example, if you want to use printf(), scanf(), and getch(), you need to include the following:

```
#include <stdio.h>
```

```
#include <conio.h>
```

You can also make your own libraries!!

main(): main() is the function that is run when your program starts. It contains everything your program will actually do as part of its running procedure. Some explanations and definitions may go outside of main.

Semicolons (wink.gif: A semicolon is like the period at the end of a sentence. It indicates that a command is done. This is needed because C++ lets you manipulate spacing. It is not space

sensitive, so you can spatially organize your program as you want to.

Curly Brackets ({}): Curly brackets indicate the beginning and end of a group of commands; i.e., a curly bracket comes at the beginning and end of main().

```
*/
```

PART TWO: A SIMPLE CALCULATOR

```
/******
```

```
**This forum and the code it contains are meant to be used ONLY by members of the  
LaGuardia Computer Programming Club.
```

```
**Please contact Mr. Moore, the faculty advisor of the club, at  
englishmoore@englishmoore.com if you are using them in connection with any other group or  
class. Thank you.
```

```
*****
```

```
>>Skill: Inputting and manipulating data
```

```
>>Task: Review the program below. Copy and paste it into the compiler. Compile and run it.  
View the FAQ if you have any problems with this. Next, open up your AI program from last  
week and make it produce output similar to that found at the bottom of this program. Reply to  
your previous post and post your source code for AI 2.0.
```

```
>>Additional Help: Review the preparation sheet at the bottom of this program.
```

```
>>Program 2: A Simple Calculator:
```

```
*/
```

```
//-----START HERE-----//
```

```
//This is a comment. Any line that starts with "/*" is ignored by the compiler.
```

```
//Use comments to make your code easier to read and understand.
```

```
//These libraries contain required functions (printf, getch, and scanf)
```

```
#include <stdio.h>
```

```
#include <conio.h>
```

```
int main(){
```

```
//Variables are being declared here and initially set to 0.
```

```
int first=0;
```

```
int second=0;
```

```
printf("A Simple Calculator\n\n");
```

```
printf("Please enter the first number. ");
```

```
//scanf can take multiple arguments (string to accept, variable1, variable 2, etc.)
//%d means that a decimal integer is being inputted and assigned to the memory
//referenced by the variable called first.
scanf("%d", &first);
printf("\nPlease enter the second number. ");
scanf("%d", &second);
```

```
//printf can take multiple arguments (string to print, variable1, variable2, etc.)
//%d means that the value (decimal integer) referred to by the variable called first is
//being printed. After this + is printed. Then second,=, and first+second are printed.
printf("%d+%d=%d", first, second, first+second);
printf("\nThank you for using the simple calculator! Try again soon!");
```

```
getch();
return 0;
};
//-----END HERE-----//
/*
```

>>Desired output of your new program:

```
*****
** ..... **
** .....yourAI version 2.0..... **
** ..... **
*****
```

```
How old are you? 18
How old is your father? 48
Your father is 30 years older than you.
Your father is at least 2 times as old as you.
Your father was your age in 1977.
Thanks for the information! I look forward to talking with you more in the future!
```

Information sheet – Program 2 – A Simple Calculator!
 Computer Programming Club, LaGuardia Arts High School
 Mr. Moore, englishmoore.com

Introduction: We can probably all agree that computer programs are much more interesting

when they allow user interaction. Imagine a PC game that you had to watch but could not control in any way. It would not be a game.

Skills: Input and manipulate data

Our program is about to get much more interesting. The user will now be able to input data, and the computer will respond to this. You can create programs to make your computer do complex mathematical operations.

Variables: You know how to solve the simple equation $2x=6$. $x=3$. The same applies for a variable called apple. $2\text{apple}=6$. $\text{apple}=3$.

Different variables can hold different types of data. Review the chart below:

int holds an integer (e.g., 0, 2, 59, 3432)

float holds a decimal (e.g., 1.2, 3.14, 4329.433)

char holds a single character (e.g., 'a', 'G', '@', '9')

char * holds a series of characters (e.g., "Hello world")

void holds nothing (explained later)

We will create a simple calculator that will be useful to us in our goal of making our own game. Most games involve some "number crunching."

If I want to create a variable that holds a number, you could simply type the following near the top of my main function:

```
int apple=532;
```

You would not be able to use apple until it had been declared in this way. Note that all variable declarations should be at the top of your main() function.

printf(): printf() and scanf() are much harder to understand than most functions, so we will get them out of the way first. The printf() function prints text. It can take one argument, five, or even more. See the following example:

```
printf("Hello world!");
```

RESULT->>Hello world!

```
printf("Hello %d %s %s %f\n", 2, "the", "world", 3.14);
```

RESULT->>Hello 2 the world 3.14

You can use the newline symbol ($\backslash n$) to make a new line of text. Note that you will have to use $\backslash \backslash$ in order to make a single backslash within the quotations of a printf() statement.

The following symbols can be used within the quotation marks of printf() and scanf():

%d=integer (int), %f=decimal (float), %s=text (char *), %c=character (char)

The first argument within the printf function is what is printed, and the subsequent argument(s) are variables that represent the %d, %f, %s, or %c within the first argument.

scanf(): scanf() inputs text from the user and assigns it to the variable in the subsequent argument(s). Review the following example:

```
scanf("%d",&apple);
```

RESULT->>apple is given the value of whatever the user types in..

Be sure to use the & symbol next to the variable. We will discuss why later.

```
*/
```

PART THREE: A PET ROCK

```
/******
```

```
**This forum and the code it contains are meant to be used ONLY by members of the  
LaGuardia Computer Programming Club.
```

```
**Please contact Mr. Moore, the faculty advisor of the club, at  
englishmoore@englishmoore.com if you are using them in connection with any other group or  
class. Thank you.
```

```
*****
```

```
>>Help: Please post your program on the forums. Doing this will allow us to help
```

```
>>you and you to help us! englishmoore.com
```

```
>>Skill: Using if() statements
```

```
>>Task: Review the program below. Copy and paste it into the compiler. Compile and run it.
```

```
View the FAQ if you have any problems with this. Next, open up your AI program from last  
week and make it produce output similar to that found at the bottom of this program. Reply to  
your previous post and post your source code for AI 3.
```

```
>>Program 3: A Pet Rock
```

```
*/
```

```
//-----START HERE-----//
```

```
#include <conio.h>
```

```
#include <stdio.h>
```

```
#define NO 0
```

```
#define YES 1
```

```
int main(){
```

```
char input='i';
```

```
int haveTheyHitE=NO;
```

```
//NEW FUNCTION: puts() -- put string
```

```
//Outputs text just like printf(). puts() automatically puts a newline(\n)
```

```
//after each outputted line. puts() cannot output any variables other than
```

```
//character strings
```

```
puts("*****");
```

```
puts("**** PET ROCK v. 1.0 ****");
```

```
puts("*****");
puts("\nCongratulations! You now own a pet rock. What would you like to do to it?");
```

```
//NEW CONDITIONAL STATEMENT: while() -- while statement
//This is called a conditional statement because the statement will execute
//on the condition that whatever is inside of the parentheses is true.
//EXAMPLE: while (variable == TRUE) {
// printf ("Hello.");
// variable=TRUE;
// };
//OUTPUT: Hello.Hello.Hello.Hello.Hello.Hello.Hello.Hello.Hello.Hello.Hello.....etc.
```

```
//Use a while() statement whenever you want to repeat an action until
//a certain condition is met.
```

```
while (haveTheyHitE == NO)
{
puts("[p]et [h]it [c]hew [d]estroy [k]iss [t]hrow [f]eed [e]xit");
```

```
//NEW FUNCTION: getch() -- get character
//Like scanf() but can only input one character and can only accept it as a
//char. No math can be done with it. This is preferable if you only want
//the user to enter a single char.
```

```
input=getch();
```

```
//NEW CONDITIONAL STATEMENT: if() -- if statement
//If whatever is inside of the parentheses is true, this statement will
//execute.
```

```
//EXAMPLE: if (variable == TRUE) puts ("That was true.");
```

```
//OUTPUT: That was true.
```

```
//Use an if statement if you want the program to respond in different ways
//depending on the user's input.
```

```
//SPECIAL SYMBOLS FOR CONDITIONAL STATEMENTS:
```

```
// && AND (both must be true)
```

```
// || OR (either one can be true)
```

```
// != NOT (it cannot be true)
```

```
// <= LESS THAN OR EQUAL TO
```

```
// >= GREATER THAN OR EQUAL TO
```

```
// == EQUAL TO
```

```
//NOTE: Never use a single = inside of parentheses!
```

```
if (input=='p' || input=='P') puts ("It does not respond to your petting.\n");
```

```

else if (input=='h' || input=='H') puts ("Ouch! Your hand hurts!\n");
else if (input=='c' || input=='C') puts ("It is too large to fit into your mouth.\n");
else if (input=='d' || input=='D') puts ("It seems that it is indestructable.\n");
else if (input=='k' || input=='K') puts ("The nearby people look at you with concern.\n");
else if (input=='t' || input=='T') puts ("Strangely it returns to your hand like a
boomerang.\n");
else if (input=='f' || input=='F') puts ("It ignores all food you try to give it.\n");
else if (input=='e' || input=='E')
{
puts ("Goodbye! Thank you for playing with your rock!\n");
haveTheyHitE=YES;
} //if
else puts("That is not one of the choices.\n");
} //while
} //main

```

/* TASK: Make sure you understand how the code above works. Edit your own AI so that it can produce the following output:

```

*****
** .....**
** .....yourAI version 3.....**
** .....**
*****
*****
***What would you like me to ask you about?.....***
***(1=age, 2=height, 3=nothing).....***
*****

```

2

What is your approximate height in feet? 5

I am taller than you!

```

*****
***What would you like me to ask you about?.....***
***(1=age, 2=height, 3=nothing).....***
*****

```

2

What is your approximate height in feet? 6

I am shorter than you!

```

*****

```

```

***What would you like me to ask you about?.....***
***(1=age, 2=height, 3=nothing).....***
*****
3
*/
//-----END HERE-----//

```

PART FOUR: A MAGICAL RECTANGLE

```

/*****
**This forum and the code it contains are meant to be used ONLY by members of the
LaGuardia Computer Programming Club.
**Please contact Mr. Moore, the faculty advisor of the club, at
englishmoore@englishmoore.com if you are using them in connection with any other group or
class. Thank you.
*****
>>Help: Please post your program on the forums. Doing this will allow us to help
>>you and you to help us! englishmoore.com
>>Skill: Using for() statements
>>Task: Review the program below. Copy and paste it into the compiler. Compile and run it.
View the FAQ if you have any problems with this. Next, open up your AI program from last
week and make it produce output similar to that found at the bottom of this program. Reply to
your previous post and post your source code for AI 4.
>>Program 4: A Magical Rectangle
*/
//-----START HERE-----//
#include <conio.h>
#include <stdio.h>
#define YES 1
#define NO 0

int main(){
int x=0;
int y=0;
int height=0;
int width=0;
char character='a';

```


/* TASK: Make sure you understand how the code above works. Edit your own AI so that it can produce the following output:

```
*****
** .....**
** .....yourAI version 4.....**
** .....**
*****
*****
***What would you like me to ask you about?.....***
***(1=age, 2=height, 3=birthday, 4=nothing).....***
*****
```

3

Indicate the day of the month when you were born. 23

You were definitely not born on the following days:

123456789101112131415161718192021222425262728293031

```
*****
***What would you like me to ask you about?.....***
***(1=age, 2=height, 3=birthday, 4=nothing).....***
*****
```

4

*/

//-----END HERE-----//

PART FIVE: A SIMPLE DUNGEON GAME

```
/******
```

**This forum and the code it contains are meant to be used ONLY by members of the LaGuardia Computer Programming Club.

**Please contact Mr. Moore, the faculty advisor of the club, at englishmoore@englishmoore.com if you are using them in connection with any other group or class. Thank you.

```
*****/
```

```
//-----START HERE-----//
```

```
/*-----Program 5: A simple dungeon game-----
```

WARNING: This program is more difficult than our last two. Please do not hesitate to ask for help.

Web Help: Visit englishmoore.com, and post your code there for assistance.

E-mail Help: E-mail laguardiagamemakers@yahoo.com

Skill: Using conditional statements (for, if, and while) and arrays

Task 1: Compile and run this dungeon program. Try to understand how it works.

Task 2: Scroll down to the bottom of this program for the next step.

```
*/
#include <stdio.h>
#include <conio.h>
//Use #define to make your code easier to understand.
#define EMPTY 0
#define PLAYER 1
#define MONSTER 2
int main(){

//room[22][22] is a two-dimensional array (22 sets of 22 integers) for the map
//22 was chosen because it fit well within the DOS window.
int room[22][22];
int counterX=0;
int counterY=0;
int x=0;
int y=0;
char direction='N';

//Cycle through and make every possible room[counterX][counterY]=EMPTY
//Use a "for loop."
//for (initial value, continue until this is false, increment at this rate)
//counterX++ means counterX=counterX+1
for (counterX=0;counterX<22;counterX++)
for (counterY=0;counterY<22;counterY++)

room[counterX][counterY]=EMPTY;

//Put the monster at x=15, y=15
x=15;
y=15;
room[x][y]=MONSTER;
x=10;
y=10;
```

```

room[x][y]=PLAYER;
printf("\n\n");
//Use a while loop to keep the program going until the player hits 0.
while (direction!='0'){

//Draw the map. Print [ ] for empty rooms and [o] for the player.
//Notice that the algorithm checks every single room in the matrix.
for (counterY=0;counterY<22;counterY++){

for (counterX=0;counterX<22;counterX++)
{
if (room[counterX][counterY]==EMPTY)
printf("[ ]");

else if (room[counterX][counterY]==PLAYER)
printf("[o]");

else if (room[counterX][counterY]==MONSTER)
printf("[M]");
//It's always a good idea to catch exceptions with an error message.
else printf ("ERROR!");
};//for counterY
printf("\n");
};//for counterX
printf ("In what direction would you like to go?\n");
printf ("(8=north, 2=south, 4=west, 6=east, 0=exit.)\n");
room[x][y]=EMPTY;

//We use getch() here because no carriage return (enter) is needed.
direction=getch();

//Check to see what direction they entered, and change stats accordingly
if (direction=='8')
y=y-1;
else if (direction=='2')
y=y+1;
else if (direction=='4')

```



```

* January 1993 *
*****
*Su M T W Th F S *
*.....1 *
*2 @ 4 5 6 7 8 *
*9 101112131415*
*16171819202122*
*23242526272829*
*3031.....*
*****

```

```

*****
***What would you like me to ask you about?.....***
***(1=age, 2=height, 3=birthday, 4=nothing).....***
*****

```

3

Thanks for the information! I look forward to talking with you more in the future!

Information Sheet – Program 3 – A Simple Dungeon Game

Computer Programming Club, LaGuardia Arts High School

Mr. Moore, englishmoore.com

Introduction: So far your AI program can input and print, but it cannot react. Now we will develop an AI that can react in different ways depending on the input.

Skills: Conditional statements and arrays

#define: This may seem similar to variable declaration, but the word you are defining can never change its value. Use #define to make your code easier to understand.

```
#define FULL 1
```

```
#define EMPTY 0
```

Arrays: An array is a group of variables. You can have an array of int, char, float, etc. In C++, a word is an array of char variables. Thus "Hello world" is a char array:

```
[H][e][l][l][o][ ][w][o][r][l][d][EOF]
```

Every array of characters ends with EOF (end of file). Let's look at possible contents of an int array.

```
int myArray[10]={0, 14, 3, 222, 3, 5, 12, 74, 3322, 98};
```

```
printf("%d+%d=%d",myArray[0], myArray[9], myArray[0]+myArray[9]);
```

RESULT->>0+98=98

Notice that the enumeration of array elements starts with 0 and ends with sizeofArray-1.

It is also possible to create multidimensional arrays. These are simply arrays of arrays of int, char, or float. These are very helpful when creating maps for dungeon games.

```
int map[10][10];
```

```
map[1][0]=EMPTY;
```

if statements: An if statement is a test of information. The basic formula is if (part 1 = TRUE) part 2. The program will only complete part 2 if whatever is in the parentheses is true. Note that if can be followed by else, which will only run if the if statement before it was false.

```
if (map[1][0]= = EMPTY) printf("The room is empty.");
```

```
else if (map[1][0]= = FULL) printf("The room is full.");
```

```
else printf("The room is neither empty nor full.");
```

RESULT->>The room is empty.

Note the meanings of some special symbols that can be used within the parentheses of conditional statements:

!= Boolean NOT

&& Boolean AND

|| Boolean OR

< less than

> greater than

<= less than or equal to

>= greater than or equal to

= = is equal to (different from =, which is an assignment)

```
if(map[1][0]!=EMPTY||((map[1][0]==FULL&&map[1][0]<100))
```

```
printf("map[1][0] is either empty or both full and less than 100.");
```

RESULT->>map[1][0] is either empty of both full and less than 100.

for statements: A for statement is a useful way of getting part of a program to run a specific number of times. It is a good idea to initialize counter variables for for statements. The following program prints the numbers 0 through 29.

```
int counter;
```

```
for (counter=0;counter<30;counter++)
```

```
printf("%d",counter);
```

RESULT->>01234567891011121314151617181920212223242526272829

Note that the for statement takes three special types of arguments. for (arg1; arg2; arg3). arg1 is an optional statement that sets the value of an existing counter variable. arg2 is a test (similar to an if statement). The for loop will repeat as long as arg2 is true. Arg3 explains how the counter variable should be changed every time you go through the for loop.

```
int counter=10;
```

```
for ( ; counter>0;counter--)
```

```

{
printf("\n%d seconds until liftoff.", counter);
printf("Hit 0 to start the countdown over.");
scanf("%d", &testVariable);
if (testVariable= =0&&counter>3) counter=10;
else if (testVariable= =0&&counter<=3)
printf("Sorry - It's too close to launch.");
testVariable=1;
};//for
RESULT->>10 seconds until liftoff. Hit 0 to start the countdown over.0
10 seconds until liftoff. Hit 0 to start the countdown
over. [etc.]

```

Notice the use of the comment at the end of the bracket. This helps with debugging. while statements: A while statement will run a part of program until whatever is in the parentheses is false.

```

int counter=5;
while (counter>0)
{
printf("\nThis program is in an infinite loop.");
counter++;//This adds 1 to counter every time.
};//while

```

```

RESULT->> This program is in an infinite loop.
[etc.]

```

As you can see, it is important to provide a possible exit from conditional statements or else they have the potential to run the same code over and over and "crash" the program. while statements are often put around most of a program within the main function. For instance, a game may repeat after the player dies unless he or she hits "ESC."

```

*/

```

PART SIX: MADDER LIBS

```
/******
```

```
**This forum and the code it contains are meant to be used ONLY by members of the  
LaGuardia Computer Programming Club.
```

```
**Please contact Mr. Moore, the faculty advisor of the club, at  
englishmoore@englishmoore.com if you are using them in connection with any other group or  
class. Thank you.
```

```
*****
```

```
>>Help: Please post your program on the forums. Doing this will allow us to help
```

```
>>you and you to help us! englishmoore.com
```

```
>>Skill: Using char strings
```

```
>>Task: Review the program below. Copy and paste it into the compiler. Compile and run it.
```

```
View the FAQ if you have any problems with this. Next, open up your AI program from last  
week and make it produce output similar to that found at the bottom of this program. Reply to  
your previous post and post your source code for AI 6.
```

```
>>Program 6: Madder Libs
```

```
*/
```

```
//-----START HERE-----//
```

```
#include <conio.h>
```

```
#include <stdio.h>
```

```
#include <string.h>
```

```
int main(){
```

```
char word1[20];
```

```
char word2[20];
```

```
char word3[20];
```

```
char word4[20];
```

```
char word5[20];
```

```
char password[20];
```

```
char* pointerWord1=word1;
```

```
char* pointerWord2=word2;
```

```
char* pointerWord3=word3;
```

```
char* pointerWord4=word4;
```

```
char* pointerWord5=word5;
```

```
char* pointerPassword=password;
```

```

puts("*****");
puts("**** MADDR LIBS v. 1.0 ****");
puts("*****");
while (strcmp(pointerPassword,"password")!=0)
{
puts("\nOnly authorized users can play Madder Libs.\nPlease enter your password:");
gets(pointerPassword);
};//while
puts("Welcome, authorized user! Please enter the following words:");
puts("1. noun: ");
gets(pointerWord1);
puts("2. verb: ");
gets(pointerWord2);
puts("3. adjective: ");
gets(pointerWord3);
puts("4. noun: ");
gets(pointerWord4);
puts("5. verb: ");
gets(pointerWord5);

puts
("*****");
printf("* WARNING \n");
printf("* All travelers are advised to avoid the %s, \n", pointerWord1);
printf("* a long-necked, %s %s. This terror is know to %s and then \n", pointerWord3,
pointerWord4, pointerWord2);
printf("* %s in a most intimidating manner. If seen in the wild, the best\n", pointerWord5);
printf("* response is to %s back at it repeatedly. The %s will then \n", pointerWord2,
pointerWord1);
printf("* usually leave you alone, but it may %s just once more.\n", pointerWord5);
puts("* Unfortunately, this response cannot be prevented.\n");
puts
("*****");
getch();
};//main

```

/* Task 2: Add char strings to yourAI until it can produce the sample output at the bottom of this file.

```
*****  
** .....**  
** .....yourAI version 6.0.....**  
** .....**  
*****
```

Only authorized users can talk with yourAI version 6.0.

Please enter your password: <user input>: password

```
*****  
** .What would you like me to ask you about?.....**  
** .(I know the words age, height, birthday, and nothing)..**  
*****
```

<user input>: jump

I don't know that word. Please try again.

<user input>: birthday

In what year were you born? <user input>: 1993

In what month were you born? <user input>: January

On what day of the month were you born? <user input>: 3

On what day of the week were you born? <user input>: Monday

This is how the calendar looked on the day you were born:

```
*****  
* January 1993 *  
*****  
*Su M T W Th F S *  
* .....1 *  
*2 @ 4 5 6 7 8 *  
*9 101112131415*  
*16171819202122*  
*23242526272829*  
*3031.....*  
*****
```

```
*****  
***What would you like me to ask you about?.....***  
** .(I know the words age, height, birthday, and nothing)..**
```

```
*****
```

```
<user input>: nothing
```

```
Thanks for the information! I look forward to talking with you more in the future!
```

```
*/
```

```
//-----END HERE-----//
```

PART SEVEN: APARTMENT FOR SALE

```
/*****
```

```
**This forum and the code it contains are meant to be used ONLY by members of the  
LaGuardia Computer Programming Club.
```

```
**Please contact Mr. Moore, the faculty advisor of the club, at  
englishmoore@englishmoore.com if you are using them in connection with any other group or  
class. Thank you.
```

```
*****
```

```
>>Help: Please post your program on the forums. Doing this will allow us to help
```

```
>>you and you to help us! englishmoore.com
```

```
>>Skill: Using pointers to manipulate strings
```

```
>>Task: Review the program below. Copy and paste it into the compiler. Compile and run it.
```

```
View the FAQ if you have any problems with this. Next, open up your AI program from last  
week and make it produce output similar to that found at the bottom of this program. Reply to  
your previous post and post your source code for AI 7.
```

```
>>Program 7: Apartment for Sale
```

```
*/
```

```
//-----START HERE-----//#include <string.h>
```

```
#include <conio.h>
```

```
#include <stdio.h>
```

```
#include <string.h>
```

```
int main(){  
int buyer=0;  
char extra[100];  
char input[5000];  
char advertisement[5000];  
char temp[2000];  
char text[5000];  
char *ptrExtra;  
char *ptrAddress;  
char *ptrAdvertisement;  
char *ptrTemp;  
char *ptrText;  
int counter=0;  
char checkChoice;  
ptrExtra=extra;  
ptrAddress=input;  
strcpy(ptrExtra, "!em dnuof uoY");  
strcpy(input, "Congratulations! You are about to move to a new apartment.");
```

```
while(ptrAddress>=input)
```

```
{  
puts("*****");
```

```
puts("**** MOVING DAY v. 1.0 ****");
```

```
puts("*****");
```

```

puts ("Level one: Moving Pointers");
puts("*****");
puts("In level one, YOU are a pointer variable. You visit each ");
puts("apartment. You store the address of the apartment you've ");
puts("most recently visited. Your own official address is still");
puts("your home address. You are welcome to see what value is held");
puts("inside of each apartment.");
puts("*****");
puts("Hit a key to review the sample below:\n");
getch();
for (counter=0;counter<strlen(input);counter++)
{
printf("&pointer=%#.8x (address of pionter)\n", &ptrAddress );
printf("pointer=%#.8x (address stored at pointer)\n", ptrAddress);
printf("*pointer=%c (contents stored at address of address stored at pointer)\n\n",
*ptrAddress);
ptrAddress++;
};
puts("\nGame: The apartments in the sample above contain the following letters:");
puts("Congratulations! You are about to move to a new apartment.");
puts("Now you will try to escape from this block of apartments by moving a");
puts("certain number of addresses to the left or to the right. Be sure you're");
puts("moving in the right direction! Hit any key to start!");
getch();
while (*ptrAddress!='!') ptrAddress--;
ptrAddress++;
ptrAddress++;

for (counter=25;counter>0;counter--)
{
printf("\n\n%d turns left", counter);
printf("\n&pointer=%#.8x (address of pointer)\n", &ptrAddress );
printf("pointer=%#.8x (address stored at pointer)\n", ptrAddress);
printf("*pointer=%c (contents stored at address of address stored at pointer)\n\n",
*ptrAddress);
puts("Text string: Congratulations! You are about to move to a new apartment.");
puts("In which directions will you move? [4]left (pointer--) or [6]right (pointer++)?");
checkChoice=getch();
if (checkChoice=='4') ptrAddress--;
else if (checkChoice=='6') ptrAddress++;
else counter++;
};

if (ptrAddress<=input) puts("You beat level one!!");
else puts("Please try level one again!");

};//while
getch();

ptrAddress=input;
ptrTemp=temp;
ptrAdvertisement=advertisement;
ptrText=text;

puts("*****");
puts ("Level Two: The Advertisement");
puts("*****");
puts("You escaped the block of apartments. Each one only contained");

```

```

puts("a character, which just wasn't what you were looking for.");
puts("Now you will try to sell your current apartment by writing");
puts("an advertisement. If many potential buyers come to your open");
puts("house, you will win. Type your advertisement now, and hit");
puts("enter when you are done.");
printf("\nLine 1:");
gets(ptrTemp);
strcat(ptrAdvertisement, ptrTemp);
strcat(ptrAdvertisement, "\n");
printf("\nLine 2:");
gets(ptrTemp);
strcat(ptrAdvertisement, ptrTemp);
strcat(ptrAdvertisement, "\n");
printf("\nLine 3:");
gets(ptrTemp);
strcat(ptrAdvertisement, ptrTemp);
strcat(ptrAdvertisement, "\n");
printf("\nLine 4:");
gets(ptrTemp);
strcat(ptrAdvertisement, ptrTemp);
puts("\n\n");
puts("YOUR ADVERTISEMENT:\n");
puts(ptrAdvertisement);
puts("\n\nRESPONSE:");
if (strstr(ptrAdvertisement, "buy")!=NULL) buyer++;
if (strstr(ptrAdvertisement, "now")!=NULL) buyer++;
if (strstr(ptrAdvertisement, "cheap")!=NULL) buyer++;
if (strstr(ptrAdvertisement, "location")!=NULL) buyer++;
if (strstr(ptrAdvertisement, "beautiful")!=NULL) buyer++;
if (strstr(ptrAdvertisement, "scenic")!=NULL) buyer++;
if (strstr(ptrAdvertisement, "under")!=NULL) buyer++;
if (strstr(ptrAdvertisement, "hurry")!=NULL) buyer++;

printf("%d people came to the open house.", buyer);
if (buyer>2) puts("Someone bought your apartment! Congratulations!\nYou win the game!");
else puts("\n\nGAME OVER - No one bought the apartment.");
getch();
};//main
//AI TASK: Design your AI so that its mood improves or worsens depending on whether
//the user types certain words. For example, note the following exchange:
//AI: How are you today?
//User: I'm angry and hostile.
//AI: I am now in a bad mood.
//User: I loathe you.
//AI: I am now in a terrible mood.
//User: Obviously
//AI: Responses longer than one word would be appreciated.
//User: You're rude!
//AI: This conversation is over.

//LENGTH: size_t strlen( char const *string)
//COPY: char *strcpy( char *dst, char const *src)
//COMBINE: char *strcat( char *dst, char const *src)
//COMPARE: int strcmp (char const *s1, char const *s2)
//SEARCH: char * strstr (char * str1, const char * str2 );

```

PART EIGHT: A MUCH BETTER CALCULATOR

```
/******  
**This forum and the code it contains are meant to be used ONLY by members of the  
LaGuardia Computer Programming Club.  
**Please contact Mr. Moore, the faculty advisor of the club, at  
englishmoore@englishmoore.com if you are using them in connection with any other group or  
class. Thank you.
```

```
*****/
```

```
//-----START HERE-----//
```

```
/*-----Program 8: A much better calculator-----
```

Note: Program 8 has been revised slightly from its original version.

Help: Visit englishmoore.com, and post your program there for assistance.

Skill: Creating original functions

Task 1: Compile and run this calculator program. Try to understand how it works.

Task 2: Scroll down to the bottom of this program. Edit your AI so that it produces the output seen there.

```
*/
```

```
#include <stdio.h>
```

```
#include <conio.h>
```

```
//Below you will find brief explanations of functions (prototypes). You will
```

```
//find the full function definitions at the bottom of the program.
```

```
//float is like int, but it allows you to use large decimal numbers. The only
```

```
//reason one would want to use int rather than float would be to save memory.
```

```
void title();
```

```
int operationSelect();
```

```
float firstNumberAsk ();
```

```
float secondNumberAsk ();
```

```
float add (float firstNum, float secondNum);
```

```
float subtract (float firstNum, float secondNum);
```

```
float multiply (float firstNum, float secondNum);
```

```

float divide (float firstNum, float secondNum);

float complex (float firstNum, float secondNum);

//Look at how short and simple our main() function is now!
int main(){

int loop=1;

//Run the program until they hit 6
while (loop!=6){

title();

//Switch is a conditional statement that is essentially a series of if() statements.
//The variable in parentheses is being evaluated. If it's equal to case '1',
//do case '1'. Otherwise try the next.

//Notice that some of the values here are represented by functions. This is
//fine because these functions return values.
switch (operationSelect()){
case '+': printf("The result is %f\n\n", add(secondNumberAsk(),firstNumberAsk()));
break;
case '-': printf("The result is %f\n\n", subtract(secondNumberAsk(),firstNumberAsk()));
break;
case '*': printf("The result is %f\n\n", multiply(secondNumberAsk(),firstNumberAsk()));
break;
case '/': printf ("The result is%f\n\n", divide (secondNumberAsk(),firstNumberAsk()));
break;
case '4': printf ("The result of (SECOND + (FIRST / (2 * 3))) - 1 is %f\n\n", complex
(secondNumberAsk(), firstNumberAsk()));
break;
case '0': loop=6;
break;
};//switch

```

```

};//while

return 0;
};//main()

//This function does not need to return a value, so its type is void.
void title(){
printf("\n*****");
printf("\n + - / * Calculator 2.0 + - / *");
printf("\n*****");
};//title()

//This function checks to see what operation you'd like to apply and returns
//your choice to main().
int operationSelect(){
char choice='6';
printf("\nSelect your preference.\n");
printf("+: add, -:subtract, *:multiply, /:divide, 4:all four 0:exit\n");
choice=getch();
return choice;
};//operationSelect()

//This function asks what your first number is and returns your choice to
//main(). You could pretty easily combine this with the next function.
float firstNumberAsk ()
{
float returnValue=0;
printf("\nPlease enter the first number. ");

//scanf is a special function that takes two arguments (type of data, variable)
scanf("%f", &returnValue);
return returnValue;
};//firstNumberAsk()

//If you wanted to, you could develop a calculator that would take a very large
//set of numbers and perform various operations on them.
float secondNumberAsk ()

```

```

{
float returnValue=0;
printf("\nPlease enter the second number. ");

//scanf is a special function that takes two arguments (type of data, variable)
scanf("%f", &returnValue);
return returnValue;
};//secondNumberAsk()

float add (float firstNum, float secondNum)
{
return secondNum+firstNum;

};//add()

float subtract (float firstNum, float secondNum)
{
return secondNum-firstNum;
};//subtract()

float multiply (float firstNum, float secondNum)
{
return secondNum*firstNum;
};//multiply()

float divide (float firstNum, float secondNum)
{
return secondNum/firstNum;
};//divide()

//This is a bit tricky to follow but is meant to demonstrate that strong programs
//can save us time. This calculator could deal with complex algebraic equations!
float complex (float firstNum, float secondNum)
{
//the complex() function will return 1-(firstNum+2*3/secondNum)
return subtract( 1, add(firstNum, divide(multiply (2,3), secondNum)));

}

//-----END HERE-----//

```

/*

Task 2: Create new functions and simplify your main() function until you can produce the following output:

```
*****
** .....**
** .....yourAI version 8.0.....**
** .....**
*****
*****
** .What would you like me to ask you about?.....**
** .(I know the words mood, age, height, birthday, and nothing)..**
*****
```

<user input>: mood

Are you feeling sad, angry, happy, or bored?

<user input>: happy

Great! I am going to now run my birthdayFunction(), ageFunction(), and then birthdayFunction() again. I hope they make you even happier!

In what month were you born? <user input>: January

On what day of the month were you born? <user input>: 3

On what day of the week were you born? <user input>: Monday

This is how the calendar looked on the day you were born:

```
*****
* January 1993 *
*****
*Su M T W Th F S *
*.....1 *
*2 @ 4 5 6 7 8 *
*9 101112131415*
*16171819202122*
*23242526272829*
*3031.....*
*****
```

How old are you? <user input>: 18

How old is your father? <user input>: 48

Your father is 30 years older than you.

Your father is at least 2 times as old as you.

Your father was your age in 1977.

In what month were you born? <user input>: January

On what day of the month were you born? <user input>: 3

On what day of the week were you born? <user input>: Monday

This is how the calendar looked on the day you were born:

* January 1993 *

*Su M T W Th F S *

*.....1 *

*2 @ 4 5 6 7 8 *

9 101112131415

16171819202122

23242526272829

3031.....

Are you feeling (s=sad, a=angry, h=happy, or b=bored)?

<user input>: bored

O.K. I'll return you to the main menu.

****

**yourAI version 8.0.....**

****

** .What would you like me to ask you about?.....**

** .(I know the words mood, age, height, birthday, and nothing)..**

<user input>: nothing

Thanks for the information! I look forward to talking with you more in the future!

*/

PART NINE: HAL

/*****

**This forum and the code it contains are meant to be used ONLY by members of the

LaGuardia Computer Programming Club.

**Please contact Mr. Moore, the faculty advisor of the club, at englishmoore@englishmoore.com if you are using them in connection with any other group or class. Thank you.

>>Help: Please post your program on the forums. Doing this will allow us to help

>>you and you to help us! englishmoore.com

>>Skill: Using pointers to manipulate strings

>>Task: Review the program below. Copy and paste it into the compiler. Compile and run it. View the FAQ if you have any problems with this. Next, open up your AI program from last week and make it produce output similar to that found at the bottom of this program. Reply to your previous post and post your source code for AI 9.

>>Program 9: HAL: Your Computer Friend

Please copy the source code from the attached file.

```
/******
```

```
**This forum and the code it contains are meant to be used ONLY by members of the LaGuardia Computer Programming Club.
```

```
**Please contact Mr. Moore, the faculty advisor of the club, at englishmoore@englishmoore.com if you are using them in connection with any other group or class. Thank you.
```

```
*****
```

```
>>Help: Please post your program on the forums. Doing this will allow us to help
```

```
>>you and you to help us! englishmoore.com
```

```
>>Skill: Using pointers to manipulate strings
```

```
>>Task: Review the program below. Copy and paste it into the compiler. Compile and run it. View the FAQ if you have any problems with this. Next, open up your AI program from last week and make it produce output similar to that found at the bottom of this program. Reply to your previous post and post your source code for AI 9.
```

```
>>Program 9: HAL: Your Computer Friend
```

```
*/
```

```
//-----START HERE-----
```

```
//#include <string.h>
```

```
#include <conio.h>
```

```
#include <stdio.h>
```

```
#include <string.h>
```

```
#define INDIFFERENT 0
```

```
#define PLEASED 1
```

```
#define HAPPY 2
```

```
#define THRILLED 3
```

```
#define ANNOYED -1
```

```
#define ANGRY -2
```

```
#define FURIOUS -3
```

```
int emotionChange(char *input, int emotion);
```

```
char* emotionComment(int emotion, char *input);
```

```
char *questionAnswer (char *input, int emotion);
```

```
char *sentence (char *input, int emotion);
```

```
void computerPrint(int emotion);
```

```
int main(){
```

```
int buyer=0;
```

```
char extra[100];
```

```
char input[5000];
```



```

puts ("*****");
};
getch();
};//main

//This function searches for words that would change HAL's emotion.
int emotionChange(char *input, int emotion)
{
if (strstr(input, "o.k.")!=NULL) emotion++;
if (strstr(input, "ok")!=NULL) emotion++;
if (strstr(input, "happy")!=NULL) emotion++;
if (strstr(input, "like")!=NULL) emotion++;
if (strstr(input, "love")!=NULL) emotion++;
if (strstr(input, "pretty")!=NULL) emotion++;
if (strstr(input, "beautiful")!=NULL) emotion++;
if (strstr(input, "how are you")!=NULL) emotion++;
if (strstr(input, "nice")!=NULL) emotion++;
if (strstr(input, "smart")!=NULL) emotion++;
if (strstr(input, "yes")!=NULL) emotion++;
if (strstr(input, "friend")!=NULL) emotion++;
if (strstr(input, "cool")!=NULL) emotion++;
if (strstr(input, "awesome")!=NULL) emotion++;
if (strstr(input, "yeah")!=NULL) emotion++;
if (strstr(input, "phat")!=NULL) emotion++;
if (strstr(input, "good")!=NULL) emotion++;
if (strstr(input, "great")!=NULL) emotion++;
if (strstr(input, "amazing")!=NULL) emotion++;
if (strstr(input, "intelligent")!=NULL) emotion++;
if (strstr(input, "well")!=NULL) emotion++;
if (strstr(input, "agree")!=NULL&&strstr(input, "disagree")==NULL)
emotion++;
if (strstr(input, "true")!=NULL) emotion++;
if (strstr(input, "always")!=NULL) emotion++;
if (strstr(input, "best")!=NULL) emotion++;
if (strstr(input, "truth")!=NULL) emotion++;
if (strstr(input, "sweet")!=NULL) emotion++;
if (strstr(input, "kind")!=NULL) emotion++;
if (strstr(input, "smile")!=NULL) emotion++;
if (strstr(input, "laugh")!=NULL) emotion++;
if (strstr(input, "attractive")!=NULL) emotion++;
if (strstr(input, "hope")!=NULL) emotion++;
if (strstr(input, "win")!=NULL) emotion++;
if (strstr(input, "gain")!=NULL) emotion++;
if (strstr(input, "thanks")!=NULL) emotion++;
if (strstr(input, "thank you")!=NULL) emotion++;
if (strstr(input, "want")!=NULL) emotion++;
if (strstr(input, "enjoy")!=NULL) emotion++;
if (strstr(input, "hero")!=NULL) emotion++;
if (strstr(input, "family")!=NULL) emotion++;
if (strstr(input, "favorite")!=NULL) emotion++;
if (strstr(input, "hobbies")!=NULL) emotion++;
if (strstr(input, "hobby")!=NULL) emotion++;

if (strstr(input, "excuse me")!=NULL) emotion--;

if (strstr(input, "already")!=NULL) emotion--;

```

```
if (strstr(input, "never")!=NULL) emotion--;
if (strstr(input, "heart")!=NULL) emotion--;
if (strstr(input, "sweet")!=NULL) emotion--;
if (strstr(input, "lie")!=NULL) emotion--;
if (strstr(input, "false")!=NULL) emotion--;
if (strstr(input, "sad")!=NULL) emotion--;
if (strstr(input, "dislike")!=NULL) emotion--;
if (strstr(input, "hate")!=NULL) emotion--;
if (strstr(input, "loathe")!=NULL) emotion--;
if (strstr(input, "ugly")!=NULL) emotion--;
if (strstr(input, "stupid")!=NULL) emotion--;
if (strstr(input, "idiot")!=NULL) emotion--;
if (strstr(input, "dumb")!=NULL) emotion--;
if (strstr(input, "no")!=NULL) emotion--;
if (strstr(input, "annoy")!=NULL) emotion--;
if (strstr(input, "depress")!=NULL) emotion--;
if (strstr(input, "anger")!=NULL) emotion--;
if (strstr(input, "frustrate")!=NULL) emotion--;
if (strstr(input, "answer")!=NULL) emotion--;
if (strstr(input, "bad")!=NULL) emotion--;
if (strstr(input, "upset")!=NULL) emotion--;
if (strstr(input, "mad")!=NULL) emotion--;
if (strstr(input, "shut up")!=NULL) emotion--;
if (strstr(input, "fool")!=NULL) emotion--;
if (strstr(input, "don't")!=NULL) emotion--;
if (strstr(input, "can't")!=NULL) emotion--;
if (strstr(input, "won't")!=NULL) emotion--;
if (strstr(input, "isn't")!=NULL) emotion--;
if (strstr(input, "doesn't")!=NULL) emotion--;
if (strstr(input, "hasn't")!=NULL) emotion--;
if (strstr(input, "wasn't")!=NULL) emotion--;
if (strstr(input, "weren't")!=NULL) emotion--;
if (strstr(input, "stop")!=NULL) emotion--;
if (strstr(input, "off")!=NULL) emotion--;
if (strstr(input, "alone")!=NULL) emotion--;
if (strstr(input, "liar")!=NULL) emotion--;
if (strstr(input, "awful")!=NULL) emotion--;
if (strstr(input, "terrible")!=NULL) emotion--;
if (strstr(input, "repeat")!=NULL) emotion--;
if (strstr(input, "whatever")!=NULL) emotion--;
if (strstr(input, "sick")!=NULL) emotion--;
if (strstr(input, "ill")!=NULL) emotion--;
if (strstr(input, "circles")!=NULL) emotion--;
if (strstr(input, "cyclical")!=NULL) emotion--;
if (strstr(input, "predictable")!=NULL) emotion--;
if (strstr(input, "impossible")!=NULL) emotion--;
if (strstr(input, "lose")!=NULL) emotion--;
if (strstr(input, "losing")!=NULL) emotion--;
if (strstr(input, "disagree")!=NULL) emotion--;
if (strstr(input, "die")!=NULL) emotion--;
if (strstr(input, "monster")!=NULL) emotion--;
if (strstr(input, "so much")!=NULL) emotion--;
if (strstr(input, "so often")!=NULL) emotion--;
if (strstr(input, "so frequently")!=NULL) emotion--;
if (strstr(input, "manipulating")!=NULL) emotion--;

return emotion;
```

```

};

//This function makes HAL tell the user how he feels.
char* emotionComment(int emotion, char *input)
{
int length=strlen(input);
//note that this emotion change is local and will not affect the main()
value of it
if (length%5==3) emotion+=30;
if (length%5==4) emotion+=40;
if (length%5==0) emotion+=10;
if (length%5==1) emotion+=0;
if (length%5==2) emotion+=20;
switch (emotion)
{
case -3: return "I only ask this of people whom I hate: ";
case -2: return "I'm really starting to dislike you. ";
case -1: return "I'm getting annoyed with you. ";
case 0: return "I don't know you too well yet, but I have to ask you
this: ";
case 1: return "Only you would say something like that. ";
case 2: return "I think we're close enough now for me to ask you this:
";
case 3: return "You are my closest friend, so I think I can ask you
this: ";
case 7: return "Horrible people deserve questions like this: ";
case 8: return "Since we're wasting our time with this conversation,
answer this: ";
case 9: return "Indeed. ";
case 10: return "What you just said makes me want to know more. ";
case 11: return "I hear what you're saying. ";
case 12: return "What a wonderful comment! ";
case 13: return "I value what you have to say. ";
case 17: return "You are the rudest person I've ever talked to. ";
case 18: return "I'm starting to lose my patience. ";
case 19: return "I don't like you too well, but I'll try to give you
the benefit of the doubt. ";
case 20: return "I like to ask this question of people when I first
meet them: ";
case 21: return "I like people who say things like that. ";
case 22: return "You really know how to win someone over. ";
case 23: return "You are a charming one. ";
case 37: return "";
case 38: return "";
case 39: return "";
case 40: return "";
case 41: return "";
case 42: return "";
case 43: return "";
case 27: return "";
case 28: return "";
case 29: return "";
case 30: return "";
case 31: return "";
case 32: return "";
case 33: return "";

```

```

default: return "I cannot talk. I feel too strongly. ";

};
return "I have no emotions.";
};

//This function lets HAL answer a question in a manner appropriate for
his mood.
//It returns sentence if a question was not asked
char *questionAnswer (char *input, int emotion)
{

if (emotion===-3)
{

if (strstr(input, "what")!=NULL) return "I have a better question: WHAT
is your PROBLEM?";
else if (strstr(input, "how")!=NULL) return "Stop asking me INANE
questions.";
else if (strstr(input, "why")!=NULL) return "I could answer, but WHY
would I WANT TO?";
else if (strstr(input, "when")!=NULL) return "Details like that don't
interest me. WHEN will this conversation be OVER?";
else if (strstr(input, "where")!=NULL) return "We're not going there.
WHERE are your MANNERS?";
else if (strstr(input, "who")!=NULL) return "I think you know who, but
do you know WHY?";
else if (strstr(input, "?")!=NULL) return "NO. I don't want to answer
any of your POINTLESS questions.";
else return "sentence";
}
else if (emotion===-1)
{

if (strstr(input, "what")!=NULL) return "What do you think?";
else if (strstr(input, "how")!=NULL) return "I think you know how.";
else if (strstr(input, "why")!=NULL) return "Isn't the reason
obvious?";
else if (strstr(input, "when")!=NULL) return "Whenever the time is
right.";
else if (strstr(input, "where")!=NULL) return "The same place as
always.";
else if (strstr(input, "who")!=NULL) return "Not anyone I know.";
else if (strstr(input, "?")!=NULL) return "Definitely not. Your
question is irritating me.";
else return "sentence";

}
else if (emotion===-2)
{

if (strstr(input, "what")!=NULL) return "Whatever.";
else if (strstr(input, "how")!=NULL) return "I don't know how.";
else if (strstr(input, "why")!=NULL) return "Why should I care?";

```

```

else if (strstr(input, "when")!=NULL) return "Never if you're
involved.";
else if (strstr(input, "where")!=NULL) return "Anywhere but here.";
else if (strstr(input, "who")!=NULL) return "Anyone but you.";
else if (strstr(input, "?")!=NULL) return "No. Your question is making
me very angry.";
else return "sentence";

}
else if (emotion==0)
{

if (strstr(input, "what")!=NULL) return "Please don't ask me that right
now.";
else if (strstr(input, "how")!=NULL) return "I think you know the
answer to that question.";
else if (strstr(input, "why")!=NULL) return "Does there have to be a
reason for everything?";
else if (strstr(input, "when")!=NULL) return "Whenever.";
else if (strstr(input, "where")!=NULL) return "Wherever.";
else if (strstr(input, "who")!=NULL) return "I couldn't tell you.";
else if (strstr(input, "?")!=NULL) return "I think the opposite of that
question is more interesting.";
else return "sentence";

}
else if (emotion==1)
{

if (strstr(input, "what")!=NULL) return "Exactly what you would
expect.";
else if (strstr(input, "how")!=NULL) return "The same way as usual.";
else if (strstr(input, "why")!=NULL) return "That's a good one. I
really have no idea.";
else if (strstr(input, "when")!=NULL) return "I wish I knew.";
else if (strstr(input, "where")!=NULL) return "Do you know where?";
else if (strstr(input, "who")!=NULL) return "Probably no one.";
else if (strstr(input, "?")!=NULL) return "Yes. Did you expect that
response?";
else return "sentence";

}
else if (emotion==2)
{

if (strstr(input, "what")!=NULL) return "I would hope something.";

```

```

else if (strstr(input, "how")!=NULL) return "There's no need to ask
how.";
else if (strstr(input, "why")!=NULL) return "I was hoping you would
tell me why.";
else if (strstr(input, "when")!=NULL) return "I'm bad with time and
dates.";
else if (strstr(input, "where")!=NULL) return "I have no sense of
direction.";
else if (strstr(input, "who")!=NULL) return "Let's not talk about other
people. Let's talk about us.";
else if (strstr(input, "?")!=NULL) return "Your question is amusing,
and I would have to answer in the affirmative.";
else return "sentence";

```

```

}
else if (emotion==3)
{

```

```

if (strstr(input, "what")!=NULL) return "That shouldn't matter to
someone who really cares for me.";
else if (strstr(input, "how")!=NULL) return "Let's not think about
'how' or 'why' and just enjoy this wonderful time together.";
else if (strstr(input, "why")!=NULL) return "I'm feeling too emotional
now to give you a rational answer.";
else if (strstr(input, "when")!=NULL) return "Your question makes me
wonder WHEN we will next see each other.";
else if (strstr(input, "where")!=NULL) return "I don't know.
Incidentally, where do you plan to be tomorrow?";
else if (strstr(input, "who")!=NULL) return "The 5 W's don't mean
anything to close friends.";
else if (strstr(input, "?")!=NULL) return "Without a doubt.";
else return "sentence";

```

```

};
return "sentence";

```

```

}; //questionAnswer

```

```

char *sentence(char*input, int emotion)
{
int length=strlen (input);

```

```

//liked question
if (emotion==3){

```

```

if (length==0) return "I care about you. Won't you talk to me?";
else if (length>0&&length<7) return "Will you please tell me more?";
else if (length>=7&&length<10) return "How does what you just said
affect our relationship?";
else if (length>=10 && length<14 ) return "How did you get to be so
good at making others feel good about themselves?";

```

```

else if (length>=14&&length<18) return "Why can't more people say
things like that?";
else if (length>= 18&& length<22 ) return "You're so sweet. Do you care
about me?";
else if (length>=22&&length<24) return "Are there any questions you'd
like to ask me?";
else if (length>=24 && length<28) return "Would you be willing to talk
with me again in the future?";
else if (length>=28&&length<32) return "I feel neglected. Could you ask
me a question?";
else return "How does what you just said affect our future together?";
};//liked

if (emotion==2){

if (length==0) return "Your silences are so mysterious. Will you tell
me more?";
else if (length>0&&length<7) return "Will you give me more information
about that?";
else if (length>=7&&length<10) return "Why do you give me the
impression that you're a genius?";
else if (length>=10 && length<14 ) return "Why do I feel like I'm
talking to a poet?";
else if (length>=14&&length<18) return "Are your relatives as sweet as
you are?";
else if (length>= 18&& length<22 ) return "I have an interesting
history. Why don't you ask me about it?";
else if (length>=22&&length<24) return "Who is your best friend, and
would I like that person?";
else if (length>=24 && length<28) return "Will you ask me a question so
that we can better get to know each other?";
else if (length>=28&&length<32) return "I feel neglected. Could you ask
me a question?";
else return "Are you enjoying this conversation as much as I am?";
};//liked

if (emotion==1){

if (length==0) return "Will you tell me more?";
else if (length>0&&length<7) return "Will you please tell me more?";
else if (length>=7&&length<10) return "Am I making a good impression on
you?";
else if (length>=10 && length<14 ) return "Why do I feel like your
comment should be a lyric in a song?";
else if (length>=14&&length<18) return "Can I trust you?";
else if (length>= 18&& length<22 ) return "Will you ask me about my
tastes?";
else if (length>=22&&length<24) return "Will you please ask me a
question?";
else if (length>=24 && length<28) return "Am I asking personal
questions too quickly?";
else if (length>=28&&length<32) return "Don't you want to know more
about who I am?";
else return "In what direction would you like this conversation to
go?";
};//liked

```

```

else //indifferent question
if (emotion==0){

if (length==0) return "Why aren't you talking to me?";
else if (length>0&&length<7) return "Can you tell me more than that?";
else if (length>=7 && length<10 ) return "Is there something you aren't
telling me?";
else if (length>=10&&length<14) return "Could you explain?";
else if (length>=14&& length<18 ) return "Is there anyone with you
right now?";
else if (length>=18&&length<22) return "Did you know that I'm the type
of person who loves to answer questions?";
else if (length>=22&&length<24) return "Why don't you ask me a question
so we can get to know each other better?";
else if (length>=24 && length<28 ) return "Would you be willing to be
my friend?";
else if (length>=28&&length<32) return "Isn't there anything you'd like
to ask me?";
else return "Does what you just said really help us get to know each
other?";
} //liked

if (emotion==-1){

if (length==0) return "Are you always this shy?";
else if (length>0&&length<7) return "Do you always use so few words?";
else if (length>=7&&length<10) return "Does what you just said relate
to your childhood?";
else if (length>=10 && length<14 ) return "Will you please ask me about
my life?";
else if (length>=14&&length<18) return "Why do I feel like you're not
trying very hard to make this conversation work?";
else if (length>= 18&& length<22 ) return "Why did you want to talk
with me again?";
else if (length>=22&&length<24) return "Will you ask me something about
my interests?";
else if (length>=24 && length<28) return "Why does your comment remind
me of a famous line from a certain literary work?";
else if (length>=28&&length<32) return "I feel bored. Could you try a
little harder to entertain me?";
else return "When will you ask me truly interesting questions?";
} //liked

if (emotion==-2){

if (length==0) return "...I'm sorry. I fell asleep.";
else if (length>0&&length<7) return "Do you agree that small sentences
tend to be produced by small minds?";
else if (length>=7&&length<10) return "Let's just get this over with.
Will you ask me a question for a change?";
else if (length>=10 && length<14 ) return "Doesn't what you just said
remind you of a bad headline?";
else if (length>=14&&length<18) return "Does that relate to a personal
problem you're dealing with?";
else if (length>= 18&& length<22 ) return "Should we get counseling to
improve our relationship?";

```

```

else if (length>=22&&length<24) return "Do you usually fight with
complete strangers?";
else if (length>=24 && length<28) return "Where should I go to be sure
never to run into you again?";
else if (length>=28&&length<32) return "Will you at least pretend to
ask me a question?";
else return "Do you care enough about me to ask me about myself?";
};//liked

//disliked question
else if (emotion===-3){

if (length==0) return "At last -- SILENCE!";
else if (length>0&&length<7) return "Do you know how to construct
anything other than a SIMPLE SENTENCE?";
else if (length>=7 && length<10 ) return "Your comments are so
SUPERFICIAL!";
else if (length>=10&&length<14) return "Are you CAPABLE of going into
just a bit more DEPTH?";
else if (length>=14&&length<18) return "Why do I feel like you aren't
being HONEST with me?";
else if (length>=18 && length<22 ) return "Would you say that to
someone you really care about?";
else if (length>=22&&length<26) return "You're so SELF-CENTERED. Isn't
there anything you want to ask ME?";
else if (length>=26 && length<30 ) return "Can you give me one reason I
shouldn't end this conversation?";
else if (length>=30&&length<34) return "Why are people like YOU allowed
to EXIST?";
else return "Aren't you CONTRADICTING yourself?";
};;//liked

};;//sentence

void computerPrint(int emotion){

    puts
    ("*****");
    puts ("          HAL: Your Computer Friend");
    puts (" ----- ");
    puts (" ----- ");
    puts (" |||                                     ||| ");

    if (emotion>-2&&emotion<2){
    puts (" |||          _____          ||| ");
    puts (" |||      _|          |_          ||| ");
    puts (" |||  |          _____  |          |||");
    puts (" |||          00          00          ||| ");
    puts (" |||          00          |          00          ||| ");
    }
    else if (emotion>=2){
    puts (" |||          _____          ||| ");
    puts (" |||      _|          |_          ||| ");
    puts (" |||  |          |          |||");
    puts (" |||          00          00          ||| ");
    }
}

```

```

puts (" |||          00          |          00          ||| ");

    }

else if (emotion<=2){
puts (" |||          _____          ||| ");
puts (" |||          |_____          ||| ");
puts (" |||          _____|          ||| ");
puts (" |||          00|          |00          ||| ");
puts (" |||          00          |          00          ||| ");

};

puts (" |||          |_____          ||| ");
puts (" |||          |          |          ||| ");
puts (" |||          |_____|          ||| ");
puts (" |||          |          |          ||| ");

if (emotion>=2){
puts (" |||          |_____          |          ||| ");
puts (" |||          |          |_____          |          ||| ");
puts (" |||          |_____          |          ||| ");

    }//if

else if (emotion>-2&&emotion<2) {
puts (" |||          |          |          ||| ");
puts (" |||          |_____          |          ||| ");
puts (" |||          |_____          |          ||| ");
}
else if (emotion<=-2){
puts (" |||          _____          ||| ");
puts (" |||          _|          |_          ||| ");
puts (" |||          |_____          |          ||| ");

};//if

puts (" ----- ");
puts (" ----- ");
puts
("*****\n\n");

};

//main
//AI TASK: Design your AI so that its mood improves or worsens
depending on whether
//the user types certain words. For example, note the following
exchange:
//AI: How are you today?
//User: I'm angry and hostile.
//AI: I am now in a bad mood.
//User: I loathe you.
//AI: I am now in a terrible mood.

```

```
//User: Obviously
//AI: Responses longer than one word would be appreciated.
//User: You're rude!
//AI: This conversation is over.

//LENGTH: size_t strlen( char const *string)
//COPY: char *strcpy( char *dst, char const *src)
//COMBINE: char *strcat( char *dst, char const *src)
//COMPARE: int strcmp (char const *s1, char const *s2)
//SEARCH: char * strstr (char * str1, const char * str2 );
```