Please write neatly and show all of your work. Adequate space for your response has been provided following each question. If you need additional room, turn to the backside of the page or ask the instructor for paper. Good luck!

<table>
<thead>
<tr>
<th>Question</th>
<th>Points Available</th>
<th>Points Awarded</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>40</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>100</td>
<td></td>
</tr>
</tbody>
</table>
1) **(20 points)** Write the screen display for the following script and associated function in the space provided.

```matlab
x = 4
y = 3
z=[2 4 5 3 1];
for i=1:2:5
    if i <= 2
        out1=confuse2(i,x,y,z)
    elseif i > 3
        confuse1(i,x,y,z)
    else
        out2=confuse3(i,y,x,z)
    end
end

function confuse1(j,a,b,c)
y1=j*3/b+a^2/2-4
end

function new2 = confuse2(j,y,x,k)
for i=1:3:5
    new2(i)=confuse3(i,x,y,k);
end
end

function w=confuse3(j,d,s,k)
switch (k(j))
    case {1,2}
        w=d+9
    case {3,4}
        w=2*s
    otherwise
        w=d+s
end
```

<table>
<thead>
<tr>
<th>Display #</th>
<th>Display</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td></td>
</tr>
</tbody>
</table>
2) **(40 Points)** The following program is one solution to your programming assignment 12. It is written with function calls from main(). The function calls (underlined and in bold) are as shown. Referring to both the function calls and the function definitions (provided without prototype), write appropriate function prototypes and definitions in the space provided.

```c
#include<stdio.h>
#include<math.h>
#define  MAXSIZE  50

// PROTOTYPES:

// enter prototype for header

// enter prototype for getxydata

// enter prototype for getangle

// enter prototype for interp

// enter prototype for displayresults

// enter prototype for goagain

//*****************************************************************************
header( )        // complete fcn definition

displayresults( )           // complete fcn definition

main() // begin main
{ // begin main
    // variable declarations
    double  lift[MAXSIZE], angle[MAXSIZE];
    int numpts;
    double  liftangle, liftest;
    // algorithm
    header();
    // get lift data
    getxydata( angle, lift, &numpts );
    // estimate lift given angle
    do
    { // begin do..while
        liftangle = getangle( );
        liftest = interp( liftangle, angle, lift, numpts );
        displayresults( liftangle, liftest );
    } while ( goagain( ) == 1 );
}

//*****************************************************************************
header( )        // complete fcn definition

displayresults( )           // complete fcn definition
```
getxydata( ) // complete fcn definition

{ // begin getxydata
  // variable declaration
  FILE *pInfile;
  // algorithm
  pInfile = fopen("liftdata.dat", "r");;
  *pnumpts = 0;
  while ( fscanf(pInfile," %lf %lf",&x[*pnumpts],&y[*pnumpts]) == 2 ){
    *pnumpts = *pnumpts+1;  } // end while
  fclose(pInfile);
} // end getxydata

getangle( ) // complete fcn definition

{ // begin getangle
  // variable declaration
  double angle;
  // algorithm
  printf( "\n\nPlease enter angle for lift estimate ==> ");
  scanf( "%lf", &angle );
  return( angle );
} // end getangle

interp( ) // complete fcn definition

{ // begin interp
  // variable declaration
  double y, slope;
  int i;
  // algorithm
  if ( x <= xvec[0] ) {
    y = yvec[0];
  } else if ( x >= xvec[npts-1] ) {
    y = yvec[npts-1];
  } else {
    i = 0;
    while ( x > xvec[i] ) {
      i = i+1;
    }  
slope = (yvec[i]-yvec[i-1])/(xvec[i]-xvec[i-1]);
    y = yvec[i]+slope*(x-xvec[i]);  }

  return(y);
} // end interp

goagain( ) // complete fcn definition

{ // begin goagain
  // variable declaration
  int choice;
  // algorithm
  printf( "\n\nAnother estimate? 1 = yes, 2 = no ==> ");
  scanf( "%d", &choice );
  return( choice );
} // end goagain} // end goagain
3) **(20 Points)** The following program is to get a list of numbers from the user and then display the list to the screen. The list must be displayed as

list[0] = first value input
list[1] = second value input
etc

The program code to get the list of numbers is provided. You are to provide the code to display the list of numbers using a function called `displaylist`. You must write the prototype, the calling statement, and the code in the spaces provided.

```
#include<stdio.h>
define MAXSIZE 20
// prototypes
    void getdata( int vec[], int *pvecsize );

main()
    { // begin main
        // variable declaration
            int list[MAXSIZE];
            int listed;
            getdata( list, &listed );

    } // end main
    //rogram code for function displaylist here
```
4) **(15 Points)** You are in the process of creating a program which performs matrix manipulations on square matrices. The first operation you want to include is matrix addition. The code for the main program is shown below. A prototype statement for a function called “addmat”, which must determine matrix C by adding matrix A to matrix B, has already been specified.

Based on the structure of the prototype statement, complete the following instructions in the spaces provided within the program:

1) Specify the correct call statement to this function
2) Write an appropriate function definition (header)
3) Write the line(s) of code which adds the corresponding matrix elements of A and B, and assigns the result to C (using the array pointer variables defined in the function).
4) Write the line(s) of code which prints each element of the resulting matrix using 10 spaces and 2 places of precision

```c
#include<stdio.h>
define N 3    //Maximum matrix dimension

// Function Prototypes
void addmat(int matdim, double inmat1[][N], double inmat2[][N], double outmat[][N]);

void main(void)
{ /* begin main */
    int dim = 3;          //identifies dimension of square matrices
    double  A[][N] = {1, 2, 3, 4, 5, 6, 7, 8, 9};   //MATRIX A
    double  B[][N] = {9, 8, 7, 6, 5, 4, 3, 2, 1};   //MATRIX B
    double  C[N][N];   //MATRIX C, result of operation chosen

    // Enter function call statement
}
/* end main */

/*/ Function "addmat" */

{ // Enter function definition
    int i, j;
    for (i=0; i<matdim; i++)
    {
        for (j = 0; j < matdim; j++)
        {

            // Enter code to add matrices

            // Enter code to print matrices

            printf("n");
        }
    }
}
```

Page 6
5) **(15 Points)** Respond to the following:

a. Write the line of code which defines a global constant named `SIZE` with a value of 10…

b. Write one line of code which declares a one dimensional array called `fractions` that contains numbers of type double, and which is initialized to have 10 elements equal to zero…

c. Write the code which assigns the value 3.333 to the seventh element of the array `fractions`…

d. Given the array `fractions` as defined and modified above, what is the output of the following piece of code (pay attention to formatting)?

```c
for (x = 5; x <= SIZE-2; x=x+1)
    printf("fractions[%d] = %lf\n", x, fractions[x]);
```

g. What prints when the following group of C statements are executed as part of a larger program. You may assume that the larger program is correctly initialized?

```c
char s1[20] = "green";
char s2[20] = "hamburgers";
char s3[15] = "and"

printf("%s\n%c%c%c\n%s\n%c%c%c\n", 
s1, s2[7], s2[6], s2[6], s2[9], s3, s2[0], s2[1], s2[2]);
```
6) (20 Points) The C program shown below is supposed to read several Fahrenheit-scale temperature values from the keyboard, convert each temperature to Celsius, and write the converted Celsius temperatures to a data file called “ctemps.dat.” The process is supposed to continue until the sentinel 9999 is entered from the keyboard.

The program contains six errors of which you are to find five. The lines are numbered 1-21 for you convenience. Circle each line number that contains an error, label it, and write the corrected version of each line at the bottom of the page.

```c
/* Fahrenheit to Celsius temperature conversion */
#include <stdio.h>

main()
{
    double F, C;
    FILE fpt;
    fpt = fopen("ctemps.dat", "r");
    /* enter first temperature */
    printf("Temperature, in degrees F: ");
    scanf("%lf", F);
    while (F != 9999) {
        C = (5.0 / 9.0) * (F - 32.0);
        fprintf("Temperature, in degrees C: %lf\n", C);
        /* enter next temperature */
        printf("Temperature, in degrees F: ");
        fscanf("%lf", &F);
    }
    printf("Bye, Have a Nice Exam!\n");
    fclose("ctemps.dat");
}
```

Corrected error 1

Corrected error 2

Corrected error 3

Corrected error 4

Corrected error 5
7) (10 points) Write the screen display for the following script and associated function in the space provided.

```c
#include <stdio.h>
void confuse4(float *pa, float *pb);
main()
{
    float a=3.2, b=5.1, temp;
    printf("a = %f  b= %f\n",a,b);
    confuse4(&a, &b);
    printf("a = %f  b= %f\n",a,b);
}
void confuse4(float *pa, float *pb)
{
    float temp;
    temp=*pa;
    *pa=*pb;
    *pb=temp;
}
```

<table>
<thead>
<tr>
<th>Display</th>
<th>Display</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>13</td>
</tr>
</tbody>
</table>