

1. Matrix multiplication $C = A*B$ can be written using two subscripts as follows:

```
for(i=0; i<n; i++)
    for(j=0; j<n; j++)
        for(k=0; k<n; k++)
            c[i][j] += a[i][k]*b[k][j];
```

Write two matrix multiplication programs: one that uses static matrix with two subscripts, and an equivalent program that uses dynamic matrices and pointer notation:

```
*(c+i*n+j) += *(a+i*n+k) * *(b+k*n+j);
```

In both cases write functions that show matrices and illustrate the work of your program using the multiplication of two 4*4 matrices.

2. Dynamic character arrays *str* and *add* contain strings. Write a function **append** that uses *str* and *add* as arguments and appends *add* to the end of *str*. Write a main program that illustrates the use of function **append** to concatenate five strings.
3. A quadratic equation $ax^2 + bx + c = 0$ has three inputs (a,b,c) and two outputs (x1 and x2). We assume that $x1 = r1 + i*t1$ and $x2 = r2 + i*t2$, $i=\text{sqrt}(-1)$.

Write an engine program for solving the quadratic equation that accepts inputs (a,b,c) as command line arguments.

If there are three command line arguments, the program should display the results (x1 and x2) on the screen.

If there are no command line arguments then the engine should prompt the user to enter a,b,c and then compute and display the results (x1 and x2) on the screen.

If there are 4 command line arguments (a b c filename.dat) then the engine should take a,b,c and store the resulting x1 and x2 in filename.dat working in silent mode (no use of screen). Write a master program that activates the engine using `system(...)` call and then retrieves results from filename.dat and displays them on the screen.