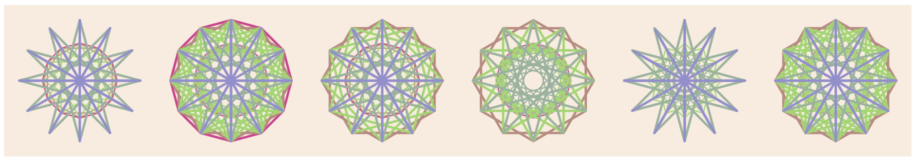
Exercises

Professor Leon Tabak
Cornell College

23 May 2022



Use these programs as a starting point for discussions and experiments.

- Copy the code by hand.

  Learning to program is like learning to play the piano. We need lots of practice. We learn through our fingers.

  A teacher can be helpful. We can learn by watching and listening to an expert, but there is no substitute for putting our own hands on the keys.

- Leave out my comments if you wish.

- Add your own comments where that is helpful.

- Give the variables names that are more meaningful to you.

- Change the values of the variables. You may make the arrays bigger or smaller.

1. Here is an easy problem to get us started.

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

int main( int argc, char** argv ) {

    printf( "Which American president spoke Chinese?\n" );

    // TO-DO: Add a statement here that prints the answer
    // to the question.

    // Hint: Search on the  Internet. This president was
    // born in West Branch, Iowa. That's less than 50 kilometers
    // from Cornell College. There is a museum there that
    // tells the story of this president's life.

    // BONUS: This president's wife also spoke Chinese.
    // The president and his wife sometimes spoke
    // Chinese in the White House.

    exit( 0 );
} // main( int, char** )
```

2. Do you think that this program produces the output that the programmer intended? Explain.

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

int main( int argc, char** argv ) {

    int a = 2;
    int b = 3;

    double average = (a + b)/2;

    printf( "average of %2d and %2d = %8.4f\n", a, b, average );

    exit( 0 );
} // main( int, char** )
```

3. What is the output of this program?

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

int main( int argc, char** argv ) {

```

```c
    int size = 5;
    int data[] = {5, 2, 3, 1, 4};

    int guess = data[0];
    for( int i = 1; i < size; i++ ) {
        if( data[i] < guess ) {
            guess = data[i];
        } // if
    } // for

    printf( "guess = %2d\n", guess );

    exit( 0 );
} // main( int, char** )
```

4. What is the output of this program?

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

int main( int argc, char** argv ) {

    int size = 5;
    int data[] = {5, 2, 3, 1, 4};

    int guess = 0;
    for( int i = 1; i < size; i++ ) {
        if( data[i] < data[guess] ) {
            guess = i;
        } // if
    } // for

    printf( "data[%2d] = %2d\n", guess, data[guess] );

    exit( 0 );
} // main( int, char** )
```

5. What is the output of this program?

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

int main( int argc, char** argv ) {

    int size = 8;
    int samples[] = { 2, 3, 5, 11, 13, 17, 19, 7 };

    for( int i = 0; i < size; i++ ) {
        printf( "%4d", samples[i] );
    } // for
    printf( "\n" );

    int i = 7;
    while( i > 0 && samples[i] < samples[i - 1] ) {
        int temp = samples[i];
        samples[i] = samples[i - 1];
        samples[i - 1] = temp;

        i--;
    } // while

    for( int i = 0; i < size; i++ ) {
        printf( "%4d", samples[i] );
    } // for
    printf( "\n" );

    exit( 0 );
} // main( int, char** )
```

6. Create a working version of the program shown below on your own computer. Study the code.

   - What is the decimal representation of the octal number 036?
   - What is the hexadecimal representation of the octal number 036?

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

int main( int argc, char** argv ) {

    // a decimal representation of an integer
    int a = 20;

    // an octal representation of an integer
    int b = 024;

    // a hexadecimal representation of an integer
    int c = 0x14;

    // print values of the 3 variables
    // as decimal ("d" for decimal) values,
    // allowing 4 digits for each value
    printf( "a = %4d\n", a );
    printf( "b = %4d\n", b );
    printf( "c = %4d\n", c );

    printf( "\n\n" );

    // print value of a in octal format
    // ( "o" for octal)
    printf( "a = %4o\n", a );

    // print value of b in hexadecimal format
    // ("x" for hexadecimal)
    printf( "b = %4x\n", b );

    // print value of c in decimal format
    // ("d" for decimal)
    printf( "c = %4d\n", c );

    exit( 0 );
} // main( int, char** )
```

7. Create a working version of the program shown below on your own computer.

   - Run the program. What do you see?
   - ˆ is one of the C programming language's bitwise operators. Which logical operation does it denote?

     (a) Is it the **bitwise complement** operator?
     (b) Is it the **bitwise and** operator?
     (c) Is it the **bitwise or** operator?
     (d) Is it the **bitwise xor** operator?
     (e) Is it the **shift left** operator?
     (f) Is it the **shift right** operator?

   - 0xFFFFFFFF is an integer expressed in hexadecimal form. (The prefix "0x" tells the compiler that what follows is a hexadecimal integer.)

     – The hexadecimal number 0xF is one less than the hexadecimal number 0x10.
     – The hexadecimal number 0xFF is one less than the hexadecimal number 0x100.
     – The hexadecimal number 0xFFF is one less than the hexadecimal number 0x1000.
     – Examine this table. Do you see a pattern?

hexadecimal	**decimal**	**hexadecimal**	**decimal**
10_{16}	$16_{10} = 16^1$	F_{16}	$15_{10} = 16^1 - 1$
100_{16}	$256_{10} = 16^2$	FF_{16}	$255_{10} = 16^2 - 1$
1000_{16}	$4096_{10} = 16^3$	FFF_{16}	$4095_{10} = 16^3 - 1$
$\vdots$	$\vdots$	$\vdots$	$\vdots$

(a) $0xFFFFFFFF$ is what power of 16 minus 1?

(b) Recall that $16^x = 2^{4x}$. For example, $16^2 = 2^8$.
$0xFFFFFFFF$ is what power of 2 minus 1?

- EXTRA CREDIT: Can you explain how this code produces these results?

  Hint: Look online to learn how to compute the two's complement of a number. Here is an explanation from Cornell University. (I teach at Cornell College, a small school in Iowa. Cornell University is a different, much larger school in New York.)

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

int main( int argc, char** argv ) {

    for( int i = 0; i < 16; i++ ) {
        int a = i;
        int b = (i ^ 0xFFFFFFFF) + 1;

        printf( "a = %d b = %d\n", a, b );
    } // for


    exit( 0 );
} // main( int, char** )
```

8. Answer the questions that you find in the comments that begin with the words "TO-DO:".

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

int main( int argc, char** argv ) {

    // populations of Iowa and neighboring states
    // TO-DO: What does 12.518e6 mean?
    float illinois = 12.518e6;
    float iowa = 3.174e6;
    float minnesota = 5.740e6;
    float missouri = 6.183e6;
    float nebraska = 1.961e6;
    float southDakota = 9.025e5;
    float wisconsin = 5.868e6;

    // array containing populations of 7 states
    float populations[] = { illinois, iowa, minnesota,
        missouri, nebraska, southDakota, wisconsin };

    // print populations
    for( int i = 0; i < 7; i++ ) {
        // TO-DO: What does "%12.0f" mean?
        printf( "%12.0f\n", populations[i] );
    } // for

    // TO-DO: Write a loop that prints the populations
    // of only those states whose population is greater
    // than 5 million people.

    // TO-DO: Write a loop that computes the sum
    // of the populations of Iowa and the six
    // states that surround Iowa.

    // TO-DO: Suggest a data structure that could hold
    // both the name and the population of each of the
    // 7 states.

    exit( 0 );
} // main( int, char** )
```