1. L1. Descrierea mediului de dezvoltare a programelor in limbajul C (TCLITE). Editare, compilare, rulare, depanare. Elemente de baza ale limbajului C. Structura unui program C.

Tip0002

```
#include <stdio.h>

void main(void)
 {
  printf("Jamsa\'s C/C++ Programmer\'s Bible!");
 }
```

Tip0004

```
#include <stdio.h>

void main(void)
 {
  printf("Jamsa\'s C/C++ Programmer\'s Bible!);
 }
```

Tip0005

```
#include <stdio.h>

void main(void)
 {
  printf("Jamsa\'s C/C++ Programmer\'s Bible!");
 }
```

Tip0006

```
#include <stdio.h>

void main(void)
 {
  printf ("Jamsa\'s ");
  printf ("C/C++ Programmer\'s ");
  printf ("Bible!");
 }
```

Tip 0007

ONE LINE

```
#include <stdio.h>

void main(void)
 {
  printf ("This is line one.");
  printf ("This is the second line.");
 }
```

TWO LINES

```
#include <stdio.h>

void main(void)
 {
  printf ("This is line one.\n");
  printf ("This is the second line.");
 }
```

Tip0008

```
#include <stdio.h>

void Main(void)
 {
  printf ("This program does not compile.");
 }
```

Tip0009

```
#include <stdio.h>

void main(void)
 {
  printf ("This is line one.");
  printf ("This is the second line.");
 }
```

Tip0012

```
#include <stdio.h>

void main(void)
 {
  print("This program does not link");
 }
```

Tip0016

```
// Program: COMMENT.C
// Written by: Kris Jamsa and Lars Klander
// Date written: 08-22-97

// Purpose: Illustrates the use of comments in a C
program.

#include <stdio.h>

void main(void)
 {
  printf ("Jamsa\'s C/C++ Programmer\'s Bible!");  //
Display a message
 }

/* Program: COMMENT.C
  Written by: Kris Jamsa and Lars Klander
  Date written: 08-22-97

  Purpose: Illustrates the use of comments in a C
program. */

#include <stdio.h>

void main(void)
 {
  printf ("Jamsa\'s C/C++ Programmer\'s Bible!");  /*
Display a message */
 }
```

Tip0017

```
#include <stdio.h>

void main(void){printf("Jamsa\'s C/C++ Programmer\'s
Bible!");}
```

Tip0018

```
void main(void)
 {
  printf ("Jamsa\'s C/C++ Programmer\'s Bible");
 }
```

Tip0020

```
#include <stdio.h>

void main(void)
 {
  // printf ("This line does not appear");

  /* This is a comment

     printf ("This line does not appear either");

  */
 }
```

2. L2. Declararea variabilelor si tipuri de baza de variabile (char, int, float, double) operatori (aritmetici si logici), instructiuni simple (atribuire, operatii artimetice si logice). Realizarea programelor ca secvente liniare de instructiuni.simple. Exemplificarea functiei printf.

Tip0081

```
#include <stdio.h>

void main(void)
 {
  int seconds_in_an_hour;
  float average;

  seconds_in_an_hour = 60 * 60;
  average = (5 + 10 + 15 + 20) / 4;
  printf("The number of seconds in an hour %d\n",
seconds_in_an_hour);
  printf("The average of 5, 10, 15, and 20 is %f\n",
average);
  printf("The number of seconds in 48 minutes is
%d\n", seconds_in_an_hour - 12 * 60);
 }
```

Tip0082

```
#include <stdio.h>

void main(void)
 {
  int remainder;
  int result;

  result = 10 / 3;
  remainder = 10 % 3;
  printf("10 Divided by 3 is %d Remainder %d\n",
result, remainder);
 }
```

Tip0085

```
#include <stdio.h>

void main(void)
 {
  int value = 0;

  while (value <= 100)
   {
    printf("%d\n", value);
    value++;
   }
 }
```

```
#include <stdio.h>

void main(void)
 {
  int value = 1;

  printf("Using postfix %d\n", value++);
  printf("Value after increment %d\n", value);
  value = 1;
  printf("Using prefix %d\n", ++value);
  printf("Value after increment %d\n", value);
 }
```

Tip0086

```
#include <stdio.h>

void main(void)
 {
  int value = 1;

  printf("Using postfix %d\n", value--);
  printf("Value after decrement %d\n", value);
  value = 1;
  printf("Using prefix %d\n", --value);
  printf("Value after decrement %d\n", value);
 }
```

Tip0087

```
#include <stdio.h>

void main(void)
 {
  printf("0 | 0 is %d\n", 0 | 0);
  printf("0 | 1 is %d\n", 0 | 1);
  printf("1 | 1 is %d\n", 1 | 1);
  printf("1 | 2 is %d\n", 1 | 2);
  printf("128 | 127 is %d\n", 128 | 127);
 }
```

Tip0088

```
#include <stdio.h>

void main(void)
 {
  printf("0 & 0 is %d\n", 0 & 0);
  printf("0 & 1 is %d\n", 0 & 1);
  printf("1 & 1 is %d\n", 1 & 1);
  printf("1 & 2 is %d\n", 1 & 2);
  printf("15 & 127 is %d\n", 15 & 127);
 }
```

Tip0089

```
#include <stdio.h>

void main(void)
 {
  printf("0 ^ 0 is %d\n", 0 ^ 0);
  printf("0 ^ 1 is %d\n", 0 ^ 1);
  printf("1 ^ 1 is %d\n", 1 ^ 1);
  printf("1 ^ 2 is %d\n", 1 ^ 2);
  printf("15 ^ 127 is %d\n", 15 ^ 127);
 }
```

Tip0090

```
#include <stdio.h>

void main(void)
 {
  int value = 0xFF;

  printf("The inverse of %X is %X\n", value, ~value);
 }
```

Tip0093

```
#include <stdio.h>

void main(void)
 {
  printf("Variables of type int use %d bytes\n",
sizeof(int));
  printf("Variables of type float use %d bytes\n",
sizeof(float));
  printf("Variables of type double use %d bytes\n",
sizeof(double));
  printf("Variables of type unsigned use %d bytes\n",
sizeof(unsigned));
  printf("Variables of type long use %d bytes\n",
sizeof(long));
 }
```

Tip0094

```
#include <stdio.h>

void main(void)
 {
  unsigned u_val = 1;
  signed int value = -1;

  printf("%u (unsigned) shifted left 2 times is %u\n",
u_val, u_val << 2);
  printf("%u (unsigned) shifted right 2 times is %u\n",
u_val, u_val >> 2);
  u_val = 65535;
  printf("%u (unsigned) shifted left 2 times is %u\n",
u_val, u_val << 2);
  printf("%u (unsigned) shifted right 2 times is %u\n",
u_val, u_val >> 2);
  printf("%d (signed) shifted left 2 times is %d\n",
value, value << 2);
  printf("%d (signed) shifted right 2 times is %d\n",
value, value >> 2);
 }
```

Tip0095

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

void main(void)
 {
   unsigned value = 1;

   printf("%u rotated right once is %u\n", value,
_rotr(value, 1));
   value = 5;
   printf("%u rotated right twice is %u\n", value,
_rotr(value, 2));
   value = 65534;
   printf("%u rotated left twice is %u\n", value,
_rotl(value, 2));
 }
```

Tip0050

```
#include <stdio.h>

void main(void)
 {
   int positive = 32767;
   int negative = -32768;

   printf("%d + 1 is %d\n", positive, positive+1);
   printf("%d - 1 is %d\n", negative, negative-1);
 }
```

Tip0051

```
#include <stdio.h>

void main(void)
 {
   float accurate = 0.123456790987654321;
   double more_accurate = 0.1234567890987654321;

   printf("Value of float\t %21.19f\n", accurate);
   printf("Value of double\t %21.19f\n",
more_accurate);
 }
```

3. L2. Continuare

Tip 0054

```
#include <stdio.h>

void main(void)
 {
  int age = 41;
  int height = 73;
  int weight = 165;

  printf("The user\'s age: %d weight: %d height: %d\n",
age, weight, height);
  printf("%d plus %d equals %d\n", 1, 2, 1 + 2);
 }
```

Tip0055

```
#include <stdio.h>

void main(void)
 {
  int value = 255;

  printf("The decimal value %d in octal is %o\n", value,
value);
  printf("The decimal value %d in hexadecimal is
%x\n", value, value);
  printf("The decimal value %d in hexadecimal is
%X\n", value, value);
 }
```

Tip0056

```
#include <stdio.h>

void main(void)
 {
  unsigned int value = 42000;

  printf("Displaying 42000 as unsigned %u\n", value);
  printf("Displaying 42000 as int %d\n", value);
 }
```

Tip 0057

```
#include <stdio.h>

void main(void)
 {
  long int one_million = 1000000;

  printf ("One million is %ld\n", one_million);
  printf ("One million is %d\n", one_million);

 }
```

Tip0058

```
#include <stdio.h>

void main(void)
 {
  float price = 525.75;
  float sales_tax = 0.06;

  printf("The item cost is %f\n", price);
  printf("Sales tax on the item is %f\n", price *
sales_tax);
 }
```

Tip0059

```
#include <stdio.h>

void main(void)
 {
  printf("The letter is %c\n", 'A');
  printf("The letter is %c\n", 65);
 }
```

Tip0060

```
#include <stdio.h>

void main(void)
 {
  float pi = 3.14159;
  float radius = 2.0031;

  printf("The circle's area is %e\n", 2 * pi * radius);
  printf("The circle's area is %E\n", 2 * pi * radius);
 }
```

Tip0061

```
#include <stdio.h>

void main(void)
 {
  printf("Displaying 0.1234 yields %g\n", 0.1234);
  printf("Displaying 0.00001234 yields %g\n",
0.00001234);
 }
```

Tip0062

```
#include <stdio.h>

void main(void)
 {
  char title[255] = "Jamsa\'s C/C++ Programmer\'s
Bible";

  printf("The name of this book is %s\n", title);
 }
```

Tip0063

```
#include <stdio.h>

void main(void)
 {
  int value;

  printf("The address of the variable value is %p\n",
&value);
 }
```

Tip0064

```
#include <stdio.h>

void main(void)
 {
  int neg_int = -5;
  int pos_int = 5;
  float neg_flt = -100.23;
  float pos_flt = 100.23;

  printf("The integer values are %+d and %+d\n",
neg_int, pos_int);
  printf("The floating point values are %+f %+f\n",
neg_flt, pos_flt);
 }
```

Tip0065

```
#include <stdio.h>

void main(void)
 {
  int value = 5;

  printf ("%1d\n", value);
  printf ("%2d\n", value);
  printf ("%3d\n", value);
  printf ("%4d\n", value);
 }
```

Tip0066

```
#include <stdio.h>

void main(void)
 {
  int value = 5;

  printf ("%01d\n", value);
  printf ("%02d\n", value);
  printf ("%03d\n", value);
  printf ("%04d\n", value);
 }
```

Tip0067

```
#include <stdio.h>

void main(void)
 {
  int value = 255;

  printf("The decimal value %d in octal is %#o\n",
value, value);
  printf("The decimal value %d in hexadecimal is
%#x\n", value, value);
  printf("The decimal value %d in hexadecimal is
%#X\n", value, value);
 }
```

Tip0068

```
#include <stdio.h>

void main(void)
 {
   float value = 1.23456;

   printf ("%8.1f\n", value);
   printf ("%8.3f\n", value);
   printf ("%8.5f\n", value);
 }
```

4. L3. Instructiuni de decizie, de selectie si de ciclare (if-else, switch-case, for, while, do-while). Instructiuni compuse. Macrodefinitii.

Tip0101

```
#include <stdio.h>

void main(void)
 {
  int age = 21;
  int height = 73;

  if (age == 21)
    printf("User\'s age is 21\n");

  if (age != 21)
    printf("User\'s age is not 21\n");

  if (height == 73)
    printf("User\'s height is 73\n");

  if (height != 73)
    printf("User\'s height is not 73\n");
 }
```

Tip0108

```
#include <stdio.h>

void main(void)
{
  int age = 10;
  int user_has_dog = 0;   // 0 is false
  if (age == 10)
{
     printf("Dogs are important pets\n");
     if (! user_has_dog)
        printf("You should get a dalmatian\n");
}
  printf("Happy is a dalmatian\n");
}
```

Tip0114

```
#include <stdio.h>

void main(void)
 {
  int counter;

  for (counter = 1; counter <= 5; counter++)
   printf("%d ", counter);

  printf("\nStarting second loop\n");

  for (counter = 1; counter <= 10; counter++)
   printf("%d ", counter);

  printf("\nStarting third loop\n");

  for (counter = 100; counter <= 5; counter++)
   printf("%d ", counter);
 }
```

Tip0116

```
#include <stdio.h>

void main(void)
 {
  int counter;

  for (counter = 5; counter >= 1; counter--)
   printf("%d ", counter);

  printf("\nStarting second loop\n");

  for (counter = 10; counter >= 1; counter--)
   printf("%d ", counter);

  printf("\nStarting third loop\n");

  for (counter = 0; counter >= 1; counter--)
   printf("%d ", counter);
 }
```

Tip0117

```
#include <stdio.h>

void main(void)
 {
  int counter;

  for (counter = -100; counter <= 100; counter += 5)
   printf("%d ", counter);

  printf("\nStarting second loop\n");

  for (counter = 100; counter >= -100; counter -= 25)
   printf("%d ", counter);
 }
```

Tip0118

```
#include <stdio.h>

void main(void)
 {
  char letter;
  float percent;

  for (letter = 'A'; letter <= 'Z'; letter++)
    putchar(letter);

  for (letter = 'z'; letter >= 'a'; letter--)
    putchar(letter);

  putchar('\n');

  for (percent = 0.0; percent < 1.0; percent += 0.1)
    printf("%3.1f\n", percent);
 }
```

Tip0120

```
#include <stdio.h>

void main(void)
 {
  int i;
  int result = 0;
  int value = 1;

  for (i = 0; i < 100; i++)
   {
    printf("%d ", i);
    result = value * --i;
   }

  printf("Result %d\n", result);
 }
```

Tip0121

```
#include <stdio.h>

void main(void)
 {
  int i, j;

  for (i = 0, j = 100; i <= 100; i++, j++)
    printf("i = %d j = %d\n", i, j);
 }
```

5. L3. Continuare

Tip0123

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

void main(void)
 {
  char letter;

          // Letter typed by the user

  printf("Do you want to continue? (Y/N): ");

  letter = getch();
                                        // Get the letter
  letter = toupper(letter);
                                        // Convert letter to
uppercase

  while ((letter != 'Y') && (letter != 'N'))
   {
    putch(7);
                        // Beep the speaker
    letter = getch();
                      // Get the letter
    letter = toupper(letter);
                                // Convert letter to
uppercase
   }

  printf("\nYour response was %c\n", letter);
 }
```

Tip0124

```
#include <stdio.h>

void main(void)
 {
  int counter = 1;  // Initialize the control variable
  while (counter <= 100)  // Test the control variable
   {
    printf("%d ", counter);  // Execute the statements
    counter++;    // Modify the control variable
   }
 }
```

Tip0125

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

void main(void)
 {
  char letter;

  do
          {
   printf("A Display directory listing\n");
   printf("B Change system time\n");
   printf("C Change system date\n");
   printf("Q Quit\n");
   printf("Choice: ");

   letter = getch();
   letter = toupper(letter);

   if (letter == 'A')
    system("DIR");
   else if (letter == 'B')
    system("TIME");
   else if (letter == 'C')
    system("DATE");
   }
  while (letter != 'Q');
 }
```

Tip0126

No_cont

```
#include <stdio.h>

void main(void)
 {
  int counter;

  printf("\nEven values\n");
  for (counter = 1; counter <= 100; counter++)
   {
    if (!(counter % 2))  // Even
     printf("%d ", counter);
   }

  printf("\nOdd values\n");
  counter = 0;
  while (counter <= 100)
   {
    counter++;

    if (counter % 2) // Odd
     printf("%d ", counter);
   }
 }
```

Odd_even

```
#include <stdio.h>

void main(void)
 {
  int counter;
  printf("\nEven values\n");
  for (counter = 1; counter <= 100; counter++)
   {
    if (counter % 2)  // Odd
     continue;
    printf("%d ", counter);
   }
  printf("\nOdd values\n");
  counter = 0;
  while (counter <= 100)
   {
    counter++;
    if (! (counter % 2)) // Even
     continue;
    printf("%d ", counter); } }
```

Tip0127

```
#include <stdio.h>

void main(void)
 {
  int counter;
  for (counter = 1; counter <= 100; counter++)
   {
    if (counter == 50)
     break;

    printf("%d ", counter);
   }

  printf("\nNext loop\n");

  for (counter = 100; counter >= 1; counter--)
   {
    if (counter == 50)
     break;

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

Tip0128

```
#include <stdio.h>

void main(void)
 {
  int count = 1;

  label:
   printf("%d ", count++);

   if (count <= 100)
    goto label;
 }
```

Tip0129

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

void main(void)
 {
  char letter;

  do {
    printf("A Display directory listing\n");
    printf("B Change system time\n");
    printf("C Change system date\n");
    printf("Q Quit\n");
    printf("Choice: ");

    letter = getch();
    letter = toupper(letter);

    switch (letter) {
      case 'A': system("DIR");
                break;
      case 'B': system("TIME");
                break;
      case 'C': system("DATE");
                break;
    };
   }
  while (letter != 'Q');
}
```

Tip0130

```
#include <stdio.h>

void main(void)
 {
   char letter;

   int vowel_count = 0;

   for (letter = 'A'; letter <= 'Z'; letter++)
     switch (letter) {
       case 'A':
       case 'E':
       case 'I':
       case 'O':
       case 'U': vowel_count++;
     };

   printf("The number of vowels is %d\n",
vowel_count);
 }
```

Tip0131

```
#include <stdio.h>

void main(void)
 {
   char letter;
   int vowel_count = 0;
   int consonant_count = 0;

   for (letter = 'A'; letter <= 'Z'; letter++)
     switch (letter) {
       case 'A':
       case 'E':
       case 'I':
       case 'O':
       case 'U': vowel_count++;
                 break;
       default: consonant_count++;
     };

   printf("The number of vowels is %d\n",
vowel_count);
   printf("The number of vowels is %d\n",
consonant_count);
 }
```

6. L4. Functii. Biblioteca de functii a limbajului C-ANSI. Operarea cu siruri (tablouri uni-dimensionale) , functii matematice din biblioteca C.

Tip0327

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

void main(void)
 {
  printf("cosine of pi/2 is %6.4f\n", cos(3.14159/2.0));
  printf("cosine of pi is %6.4f\n", cos(3.14159));
 }
```

Tip0329

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

void main(void)
 {
  double radians;

  for (radians = 0.0; radians < 3.1; radians += 0.1)
    printf("Sine of %f is %f\n", radians, sin(radians));
 }
```

Tip0331

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

void main(void)
 {
  double pi = 3.14159265;

  printf("Tangent of pi is %f\n", tan(pi));
  printf("Tangent of pi/4 is %f\n", tan(pi / 4.0));
 }
```

Tip0333

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

void main(void)
 {
  div_t result;

  result = div(11, 3);
  printf("11 divided by 3 is %d Remainder %d\n",
    result.quot, result.rem);
 }
```

Tip0334

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

void main(void)
 {
  double value;

  for (value = 0.0; value <= 1.0; value += 0.1)
    printf("exp(%f) is %f\n", value, exp(value));
 }
```

Tip0335

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

void main(void)
 {
  float value;

  for (value = -1.0; value <= 1.0; value += 0.1)
    printf("Value %f fabs %f\n", value, fabs(value));
 }
```

Tip0336

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

void main(void)
 {
  double numerator = 10.0;
  double denominator = 3.0;

  printf("fmod(10, 3) is %f\n", fmod(numerator,
   denominator));
 }
```

Tip0337

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

void main(void)
 {
  double value = 1.2345;

  double mantissa;
  int exponent;

  mantissa = frexp(value, &exponent);

  printf("Mantissa %f Exponent %d Value %f\n",
    mantissa, exponent, mantissa * pow(2.0, 1.0 *
exponent));
 }
```

Tip0338

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

void main(void)
 {
  printf("3 * 2 raised to the 4 is %f\n",
   ldexp(3.0, 4));
 }
```

Tip0339

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

void main(void)
 {
  printf("Natural log of 256.0 is %f\n", log(256.0));
 }
```

Tip0340

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

void main(void)
 {
  printf("Log10 of 100 is %f\n", log10(100.0));
  printf("Log10 of 10000 is %f\n", log10(10000.0));
 }
```

Tip0341

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

void main(void)
 {
  printf("Maximum of %f and %f is %f\n",
    10.0, 25.0, max(10.0, 25.0));
  printf("Minimum of %f and %f is %f\n",
    10.0, 25.0, min(10.0, 25.0));
 }
```

Tip0342

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

void main(void)
 {
  double value = 1.2345;
  double int_part;
  double fraction;

  fraction = modf(value, &int_part);

  printf("Value %f Integer part %f Fraction %f\n",
    value, int_part, fraction);
 }
```

Tip0343

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

void main(void)
 {
  int power;

  for (power = -2; power <= 2; power++)
    printf("10 raised to %d is %f\n", power,
      pow(10.0, power));
 }
```

Tip0344

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

void main(void)
 {
  printf("10 raised to -1 is %f\n", pow10(-1));
  printf("10 raised to 0 is %f\n", pow10(0));
  printf("10 raised to 1 is %f\n", pow10(1));
  printf("10 raised to 2 is %f\n", pow10(2));
}
```

Tip0345

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

void main(void)
 {
  int i;

  printf("Values from rand\n");
  for (i = 0; i < 100; i++)
    printf("%d ", rand());

  printf("Values from random(100))\n");
  for (i = 0; i < 100; i++)
    printf("%d ", random(100));
 }
```

Tip0346

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

void main(void)
 {
  int i;

  printf("Values from random\n");
  for (i = 0; i < 10; i++)
    printf("%f\n", random(100/100);

  printf("Values from random(-5) to random(5)/n");
  for (i = 0; i < 100; i++)
    printf("%d\n", random(10)-5);
 }
```

Tip0347

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

void main(void)
 {
  int i;

  srand(100);
  printf("Values from rand\n");
  for (i = 0; i < 5; i++)
    printf("%d ", rand());

  printf("\nSame 5 numbers\n");
  srand(100);
  for (i = 0; i < 5; i++)
    printf("%d ", rand());

  randomize();
  printf("\nDifferent 5 numbers\n");
  for (i = 0; i < 5; i++)
    printf("%d ", rand());
 }
```

Tip0348

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

void main(void)
 {
   double value;

   for (value = 0.0; value < 10.0; value += 0.1)
     printf("Value %f sqrt %f\n", value, sqrt(value));
 }
```

7. L4 Continuare

Tip0163

```
#include <stdio.h>

void main(void)
 {
  char string[256];
  int i;

  for (i = 0; i < 26; i++)
    string[i] = 'A' + i;

  string[10] = NULL;

  printf ("The string contains %s\n", string);
 }
```

```
#include <stdio.h>

void main(void)
 {
  char string[256];
  int i;

  for (i = 0; i < 26; i++)
    string[i] = 'A' + i;

  string[i] = NULL;

  printf ("The string contains %s\n", string);
 }
```

Tip0165

```
#include <stdio.h>

void main(void)
 {
  char string[] = "\"Stop!\", he said.";

  printf(string);
 }
```

Tip0166

```
#include <stdio.h>

void main(void)
 {
  char string[256];  // String input by user
  int i;           // Index into the string

  printf("Type a string of characters and press
Enter:\n");
  gets(string);

  // Display each string character until NULL is found
  for (i = 0; string[i] != NULL; i++)
    putchar(string[i]);

  printf("\nThe number of characters in the string is
%d\n", i);
 }
```

Tip0167

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

void main(void)
 {
  char book_title[] = "Jamsa\'s C/C++ Programmer\'s
Bible";

  printf("%s contains %d characters\n", book_title,
strlen(book_title));
 }
```

Tip0168

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

void main(void)
 {
  char title[] = "Jamsa\'s C/C++ Programmer\'s Bible";
  char book[128];

  strcpy(book, title);
  printf("Book name %s\n", book);
 }
```

Tip0169

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

void main(void)
 {
  char name[64] = "Triggerhill\'s I'm so";
  strcat(name, " Happy");
  printf("Happy\'s full name is %s\n", name);
 }
```

Tip0170

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

void main(void)
 {
  char name[64] = "Bill";

  strncat(name, " and Hillary", 4);
  printf("Did you vote for %s?\n", name);
 }
```

Tip0173

```
#include <stdio.h>

int streql(char *str1, char *str2)
 {
  while ((*str1 == *str2) && (*str1))
   {
    str1++;
    str2++;
   }

  return((*str1 == NULL) && (*str2 == NULL));
 }

void main(void)
 {
  printf("Testing Abc and Abc %d\n", streql("Abc",
"Abc"));
  printf("Testing abc and Abc %d\n", streql("abc",
"Abc"));
  printf("Testing abcd and abc %d\n", streql("abcd",
"abc"));
 }
```

Tip0174

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

int strieql(char *str1, char *str2)
 {
  while ((toupper(*str1) == toupper(*str2)) && (*str1))
   {
    str1++;
    str2++;
   }

  return((*str1 == NULL) && (*str2 == NULL));
 }

void main(void)
 {
  printf("Testing Abc and Abc %d\n", strieql("Abc",
"Abc"));
  printf("Testing abc and Abc %d\n", strieql("abc",
"Abc"));
  printf("Testing abcd and abc %d\n", strieql("abcd",
"abc"));
 }
```

Tip0175

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

void main(void)
 {
  printf(strlwr("Jamsa\'s C/C++ Programmer\'s
Bible!\n"));
  printf(strupr("Jamsa\'s C/C++ Programmer\'s
Bible!\n"));
 }
```

Tip0176

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

void main(void)
 {
  char title[64] = "Jamsa\'s C/C++ Programmer\'s
Bible!";
  char *ptr;

  ptr = strchr(title, 'C');
   if (*ptr)
   printf("First occurrence of C is at offset %d\n",
     ptr - title);
  else
   printf("Character not found\n");
 }
```

Tip0185

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

void main(void)
 {
  printf("Comparing Abc with Abc %d\n",
strcmp("Abc", "Abc"));
  printf("Comparing abc with Abc %d\n", strcmp("abc",
"Abc"));
  printf("Comparing abcd with abc %d\n",
strcmp("abcd", "abc"));
  printf("Comparing Abc with Abcd %d\n",
strcmp("Abc", "Abcd"));
  printf("Comparing abcd with abce %d\n",
strcmp("abcd", "abce"));
  printf("Comparing Abce with Abcd %d\n",
strcmp("Abce", "Abcd"));
 }
```

Tip0186

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

void main(void)
 {
  printf("Comparing 3 letters Abc with Abc %d\n",
   strncmp("Abc", "Abc", 3));
  printf("Comparing 3 letters abc with Abc %d\n",
   strncmp("abc", "Abc", 3));
  printf("Comparing 3 letters abcd with abc %d\n",
   strncmp("abcd", "abc", 3));
  printf("Comparing 5 letters Abc with Abcd %d\n",
   strncmp("Abc", "Abcd", 5));
  printf("Comparing 4 letters abcd with abcd %d\n",
   strncmp("abcd", "abcd", 4));
 }
```

Tip0187

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

void main(void)
 {
  printf("Comparing Abc with Abc %d\n",
   stricmp("Abc", "Abc"));
  printf("Comparing abc with Abc %d\n",
   stricmp("abc", "Abc"));
  printf("Comparing 3 letters abcd with ABC %d\n",
   strncmpi("abcd", "ABC", 3));
  printf("Comparing 5 letters abc with Abcd %d\n",
   strncmpi("abc", "Abcd", 5));
 }
```

Tip0188

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

void main(void)
 {
   int int_result;
   float float_result;
   long long_result;

   int_result = atoi("1234");
   float_result = atof("12345.678");
   long_result = atol("1234567L");

   printf("%d %f %ld\n", int_result, float_result,
long_result);
 }
```

Tip0190

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

void main(void)
 {
   printf("Searching for Abc in AbcDef %d\n",
     strspn("AbcDef", "Abc"));

   printf("Searching for cbA in AbcDef %d\n",
     strspn("AbcDef", "cbA"));

   printf("Searching for Def in AbcAbc %d\n",
     strspn("AbcAbc", "Def"));
 }
```

Tip0198

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

void main (void)
 {
   int int_value;
   float flt_value;
   long long_value;

   int_value = atoi("12345");
   flt_value = atof("33.45");
   long_value =atol("12BAD");

   printf("int %d float %5.2f long %ld\n", int_value,
     flt_value, long_value);
 }
```

8. L5. Operatii de intrare - iesire (tastatura, fisiere).

Tip0362

```
#include <stdio.h>

void main(void)
 {
  FILE *input, *output;
  int letter;

  if ((input = fopen("\\CONFIG.SYS", "r")) == NULL)
    printf("Error opening \\CONFIG.SYS\n");
  else if ((output = fopen("\\CONFIG.TST", "w")) ==
NULL)
    printf("Error opening \\CONFIG.TST\n");
  else
    {
     // Read and write each character in the file
     while ((letter = fgetc(input)) != EOF)
       fputc(letter, output);
     fclose(input);     // Close the input file
     fclose(output);    // Close the output file
    }
 }
```

Tip0286

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

void main(void)
 {
  int letter;

  printf("Type Y or N to continue and press Enter\n");

  do
  {
   letter = toupper(getchar());
  }
  while ((letter != 'Y') && (letter != 'N'));
  printf("You typed %c\n", ((letter == 'Y') ? 'Y': 'N'));
 }
```

Tip0287

```
#include <stdio.h>

void main(void)
 {
  int letter;

  for (letter = 'A'; letter <= 'Z'; letter++)
    putchar(letter);
 }
```

Tip0288

```
#include <stdio.h>

void main(void)
 {
  int letter;

  do {
    letter = getchar();
    putchar(letter);
  } while (letter != '\n');
 }
```

Tip0289

```
#include <stdio.h>

void main(void)
 {
  char string[128];

  int index = 0;
  int letter;

  printf("Type in a string and press Enter\n");
  while ((letter = getchar()) != '\n')
    string[index++] = letter;

  string[index] = NULL;

  printf("The string was: %s\n", string);
 }
```

Tip0292

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

void main(void)
 {
  int letter;

  printf("Do you want to continue? (Y/N): ");
  do
   {
    letter = getche();
    letter = toupper(letter);
   }
  while ((letter != 'Y') && (letter != 'N'));

  if (letter == 'Y')
   printf("\nYour response was Yes\n");
  else
   printf("\nWhy not?\n");
 }
```

Tip0293

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

void main(void)
 {
  int letter;

  printf("Type in a string of characters and press
Enter\n");

  do {
   letter = getch();
   letter = toupper(letter);
   putch(letter);
  } while (letter != '\r');
 }
```

Tip0295

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

void main(void)
 {
  int letter;
  int count;

  time_t start_time, stop_time;

  time(&start_time);
  for (count = 0; count < 1000; count++)
   for (letter = 'A'; letter <= 'Z'; letter++)
    putchar(letter);
  time(&stop_time);

  printf("\n\nTime required for putchar %d
seconds\n",
    stop_time-start_time);
  printf("Press any key...\n");
  getch();

  time(&start_time);
  for (count = 0; count < 1000; count++)
   for (letter = 'A'; letter <= 'Z'; letter++)
    putch(letter);
  time(&stop_time);

  printf("\n\nTime required for putch %d seconds\n",
    stop_time-start_time);
 }
```

Tip0296

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

void main(void)
 {
  int letter;
  int done = 0;
  int uppercase_found = 0;

  do {
   letter = getch();

   if (islower(letter))
    putchar(letter);
   else
    {
     if (isupper(letter))
      {
       ungetch(letter);
       uppercase_found = 1;
       putchar('\n');
      }
     done = 1;
    }
  } while (! done);

  if (uppercase_found)
   do {
    letter = getch();
    putchar(letter);
   } while (letter != '\r');
}
```

Tip0297

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

void main(void)
 {
  int count;

  time_t start_time, stop_time;

  time(&start_time);
  for (count = 0; count < 1001; count++)
   printf("Jamsa\'s C/C++ Programmer\'s Bible\n");
  time(&stop_time);

  printf("\n\nTime required for printf %d seconds\n",
   stop_time-start_time);
  printf("Press any key...\n");
  getch();

  time(&start_time);
  for (count = 0; count < 1001; count++)
   cprintf("Jamsa\'s C/C++ Programmer\'s Bible\r\n");

  time(&stop_time);

  printf("\n\nTime required for cprintf %d seconds\n",
   stop_time-start_time);
 }
```

Tip0298

```
#include <conio.h>

void main(void)
 {
  int a, b, c;

  cprintf("Type 3 integer values and press Enter\r\n");
  cscanf("%d %d %d", &a, &b, &c);
  cprintf("The values entered were %d %d %d\r\n", a, b,
c);
 }
```

Tip0299

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

void main(void)
 {
  int count;

  time_t start_time, stop_time;

  time(&start_time);
  for (count = 0; count < 1001; count++)
   printf("Jamsa\'s C/C++ Programmer\'s Bible\n");
  time(&stop_time);

  printf("\n\nTime required for printf %d seconds\n",
   stop_time-start_time);
  printf("Press any key...\n");
  getch();

  time(&start_time);
  for (count = 0; count < 1001; count++)
   puts("Jamsa\'s C/C++ Programmer\'s Bible");

  time(&stop_time);

  printf("\n\nTime required for puts %d seconds\n",
   stop_time-start_time);
 }
```

Tip0300

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

void main(void)
 {
  int count;

  time_t start_time, stop_time;

  time(&start_time);
  for (count = 0; count < 1500; count++)
   puts("Jamsa\'s C/C++ Programmer\'s Bible");
  time(&stop_time);

  printf("\n\nTime required for puts %d seconds\n",
   stop_time-start_time);
  printf("Press any key...\n");
  getch();

  time(&start_time);
  for (count = 0; count < 1500; count++)
   cputs("Jamsa\'s C/C++ Programmer\'s Bible\r\n");

  time(&stop_time);

  printf("\n\nTime required for cputs %d seconds\n",
   stop_time-start_time);
 }
```

9. L5. Continuare

Tip0301

```
#include <stdio.h>

void main(void)
 {
  char string[256];

  printf("Type in a string of characters and press
Enter\n");
  gets(string);

  printf("The string was %s\n", string);
 }
```

Tip0302

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

void main(void)
 {
  char buffer[256];

  buffer[0] = 253;  // Number of characters that can be
read

  printf("Type in a string and press Enter\n");
  cgets(buffer);

  printf("\n\nThe number of characters read was
%d\n",
    buffer[1]);

  printf("The string read: %s\n", &buffer[2]);
 }
```

Tip0304

```
#include <conio.h>

void main(void)
 {
  clrscr();
 }
```

Tip0306

```
#include <conio.h>

void main(void)
 {
  int line;

  clrscr();

  for (line = 1; line < 25; line++)
   cprintf("This is line %d\r\n", line);

  cprintf("Press a key to Continue: ");
  getch();

  gotoxy(1, 12);

  for (line = 12; line < 15; line++)
    delline();

  gotoxy(1, 25);
 }
```

Tip0307

```
#include <conio.h>

void main(void)
 {
  clrscr();

  gotoxy(1, 5);
  cprintf("Output at row 5 column 1\n");

  gotoxy(20, 10);
  cprintf("Output at row 10 column 20\n");
 }
```

Tip0308

```
#include <conio.h>

void main(void)
 {
  int row, column;

  clrscr();

  cprintf("This is line 1\r\n");
  cprintf("Line 2 is a little longer\r\n");
  cprintf("This is the last line");

  row = wherey();
  column = wherex();

  cprintf("\r\nThe cursor position was row %d column
%d\n",
    row, column);
 }
```

Tip0309

```
#include <conio.h>

void main(void)
 {
  int line;

  clrscr();

  for (line = 1; line < 25; line++)
   cprintf("This is line %d\r\n", line);

  cprintf("Press a key to Continue: ");
  getch();

  gotoxy(1, 12);

  insline();
  cprintf("This is new text!!!");
  gotoxy(1, 25);
 }
```

Tip0310

```
#include <conio.h>
#include <io.h>
#include <fcntl.h>
#include <sys\stat.h>

void main(void)
 {
  char buffer[8000];
  int handle;

  if ((handle = creat("SAVESCR.DAT", S_IWRITE)) == -1)
   cprintf("Error opening SAVESCRN.DAT\r\n");
  else
   {
    gettext(1, 1, 80, 25, buffer);
    write(handle, buffer, sizeof(buffer));
    close(handle);
   }
 }
```

Tip0311

```
#include <conio.h>
#include <io.h>
#include <fcntl.h>
#include <sys\stat.h>
#include <stdlib.h>
#include <dos.h>

void main(void)
 {
  char buffer[128];
  int row, column;

  clrscr();
  cprintf("Jamsa\'s C/C++ Programmer\'s Bible\r\n");
  gettext(1, 1, 23, 1, buffer);

  while (! kbhit())
   {
    clrscr();
    row = 1 + random(24);
    column = 1 + random(58);
    puttext(column, row, column+22, row, buffer);
    delay(2000);
   }
 }
```

Tip0313

```
#include <conio.h>

void main(void)
 {
  int color;

  for (color = 1; color < 16; color++)
   {
    textattr(color);
    cprintf("This is color %d\r\n", color);
   }

  textattr(128 + 15);
  cprintf("This is blinking\r\n");
 }
```

Tip0314

```
#include <conio.h>

void main(void)
 {
   union TextColor {
     struct {
      unsigned char foreground:4;
      unsigned char background:3;
      unsigned char blinking:1;
     } color_bits;
     unsigned char value;
   } colors;
   colors.color_bits.foreground = BLUE;
   colors.color_bits.background = RED;
   colors.color_bits.blinking = 1;

   textattr(colors.value);

   clrscr();
   cprintf("This is the new text colors\n");
 }
```

Tip0315

```
#include <conio.h>

void main(void)
 {
  int color;

  for (color = 1; color < 16; color++)
   {
     textcolor(color);
     cprintf("This is color %d\r\n", color);
   }

  textcolor(128 + 15);
  cprintf("This is blinking\r\n");
 }
```

Tip0316

```
#include <conio.h>

void main(void)
 {
   int color;

   for (color = 0; color < 8; color++)
    {
     textbackground(color);
     cprintf("This is color %d\r\n", color);
     cprintf("Press any key to continue\r\n");
     getch();
    }
 }
```

10. L6. Matrice (tablouri multidimensionale), pointeri si structure

Tip0456

```
#include <stdio.h>

void main(void)
 {
  int scores[100];
  float salaries[100];
  char string[100];

  printf("Bytes used to hold int scores[100] is %d
bytes\n",
    sizeof(scores));

  printf("Bytes used to hold int salaries[100] is %d
bytes\n",
    sizeof(salaries));

  printf("Bytes used to hold char string[100] is %d
bytes\n",
    sizeof(string));
 }
```

Tip0458

```
#include <stdio.h>

void main(void)
 {
  int scores[5] = {80, 70, 90, 85, 80};

  printf("Array Values\n");
  printf("scores[0] %d\n", scores[0]);
  printf("scores[1] %d\n", scores[1]);
  printf("scores[2] %d\n", scores[2]);
  printf("scores[3] %d\n", scores[3]);
  printf("scores[4] %d\n", scores[4]);
 }
```

Tip0459

```
#include <stdio.h>

void main(void)
 {
  int scores[5] = {80, 70, 90, 85, 80};
  int i;

  printf("Array Values\n");

  for (i = 0; i < 5; i++)
    printf("scores[%d] %d\n", i, scores[i]);
 }
```

Tip0460

```
#include <stdio.h>

#define ARRAY_SIZE 5

void main(void)
 {
  int values[ARRAY_SIZE] = {80, 70, 90, 85, 80};
  int i;

  for (i = 0; i < ARRAY_SIZE; i++)
    printf("values[%d] %d\n", i, values[i]);
 }

#include <stdio.h>

void main(void)
 {
  int values[5] = {80, 70, 90, 85, 80};
  int i;

  for (i = 0; i < 5; i++)
    printf("values[%d] %d\n", i, values[i]);
 }
```

Tip0461

```
#include <stdio.h>

void show_array(int values[], int number_of_elements)
 {
  int i;

  for (i = 0; i < number_of_elements; i++)
   printf("%d\n", values[i]);
 }

void main(void)
 {
  int scores[5] = {70, 80, 90, 100, 90};

  show_array(scores, 5);
 }
```

Tip0462

```
#include <stdio.h>

void show_array(int values[], int number_of_elements)
 {
  int i;

  printf("About to display %d values\n",
number_of_elements);
  for (i = 0; i < number_of_elements; i++)
   printf("%d\n", values[i]);
 }

void main(void)
 {
  int scores[5] = {70, 80, 90, 100, 90};
  int count[10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
  int small[2] = {-33, -44};

  show_array(scores, 5);
  show_array(count, 10);
  show_array(small, 2);
 }
```

Tip0465

```
void main(void)
 {
  char string[66000L];   // 66,000 bytes

  int values[33000L];    // 33,000 * 2 = 66,000 bytes

  float numbers[17000];  // 17,000 * 4 = 68,000 bytes
 }
```

Tip0466

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

void main(void)
 {
   int i;

   float huge *values;

   if ((values = (float huge *) halloc (17000,
sizeof(float))) == NULL)
     printf ("Error allocating huge array\n");
   else
    {
          printf("Filling the array\n");

          for (i = 0; i < 17000; i++)
           values[i] = i * 1.0;

          for (i = 0; i < 17000; i++)
           printf ("%8.1f ", values[i]);

          hfree(values);
    }
 }
```

Tip0472

```
#include <stdio.h>

void main(void)
 {
  int box[3][3];

  float year_sales[52][5];

  char pages[40][60][20];

  printf("Bytes to hold int box[3][3] %d bytes\n",
sizeof(box));
  printf("Bytes to hold float year_sales[52][5] %d
bytes\n",
    sizeof(year_sales));
  printf("Bytes to hold char pages[40][60][20] %ld
bytes\n",
    sizeof(pages));
 }
```

Tip0473

```
#include <stdio.h>

void main(void)
 {
  int row, column;

  float table[3][5] = {{1.0, 2.0, 3.0, 4.0, 5.0},
                       {6.0, 7.0, 8.0, 9.0, 10.0},
                       {11.0, 12.0, 13.0, 14.0, 15.0}};

  for (row = 0; row < 3; row++)
   for (column = 0; column < 5; column++)
    printf("table[%d][%d] = %f\n", row, column,
table[row][column]);
 }
```

Tip0474

```
#include <stdio.h>

void main(void)
 {
  int row, column, table;
  float values[2][3][5] = {
                       {{1.0, 2.0, 3.0, 4.0, 5.0},
                        {6.0, 7.0, 8.0, 9.0, 10.0},
                        {11.0, 12.0, 13.0, 14.0, 15.0}},

                       {{16.0, 17.0, 18.0, 19.0, 20.0},
                        {21.0, 22.0, 23.0, 24.0, 25.0},
                        {26.0, 27.0, 28.0, 29.0, 30.0}}
                       };

  for (row = 0; row < 2; row++)
   for (column = 0; column < 3; column++)
    for (table = 0; table < 5; table++)
     printf("values[%d][%d][%d] = %f\n", row, column,
table,
      values[row][column][table]);
 }
```

Tip0476

```
#include <stdio.h>

void show_2d_array(int array[][10], int rows)
 {
  int i, j;
  for (i = 0; i < rows; i++)
   for (j = 0; j < 10; j++)
    printf("array[%d][%d] = %d\n", i, j, array[i][j]);
 }

void main(void)
 {
  int a[1][10] = {{1, 2, 3, 4, 5, 6, 7, 8, 9, 10}};
  int b[2][10] = {{1, 2, 3, 4, 5, 6, 7, 8, 9, 10},
                  {11, 12, 13, 14, 15, 16, 17, 18, 19, 20}};
  int c[3][10] = {{1, 2, 3, 4, 5, 6, 7, 8, 9, 10},
                  {11, 12, 13, 14, 15, 16, 17, 18, 19, 20},
                  {21, 22, 23, 24, 25, 26, 27, 28, 29, 30}};

  show_2d_array(a, 1);
  show_2d_array(b, 2);
  show_2d_array(c, 3);   }
```

Tip0477

```
#include <stdio.h>

long sum_array(int array[], int elements)
 {
  long sum = 0;

  int i;

  for (i = 0; i < elements; i++)
   sum += array[i];

  return(sum);
 }


void main(void)
 {
  int a[10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
  int b[2][10] = {{1, 2, 3, 4, 5, 6, 7, 8, 9, 10},
                  {11, 12, 13, 14, 15, 16, 17, 18, 19, 20}};
  int c[3][10] = {{1, 2, 3, 4, 5, 6, 7, 8, 9, 10},
                  {11, 12, 13, 14, 15, 16, 17, 18, 19, 20},
                  {21, 22, 23, 24, 25, 26, 27, 28, 29, 30}};

  printf("Sum of first array elements %d\n",
sum_array(a, 10));
  printf("Sum of second array elements %d\n",
sum_array(b, 20));
  printf("Sum of third array elements %d\n",
sum_array(c, 30));
 }
```

Tip0482

```
#include <stdio.h>

void main(void)
 {
  union EmployeeDates
            {
   int days_worked;
   struct Date
                     {
    int month;
    int day;
    int year;
   } last_day;
  } emp_info;

  union Numbers
            {
   int a;
   float b;
   long c;
   double d;  // Largest--requires 8 bytes
  } value;

  printf("Size of EmployeeDates %d bytes\n",
sizeof(emp_info));
  printf("Size of Numbers %d bytes\n", sizeof(value));
 }
```

11. L6. Continuare

Tip0508

```
#include <stdio.h>

void main(void)
 {
  int count = 1;
  float salary = 40000.0;
  long distance = 1234567L;

  printf("Address of count is %x\n", &count);
  printf("Address of salary is %x\n", &salary);
  printf("Address of distance is %x\n", &distance);
 }
```

Tip0509

```
#include <stdio.h>

void main(void)
 {
  int count[10];
  float salaries[5];
  long distances[10];

  printf("Address of the array count is %x\n", count);
  printf("Address of the array salaries is %x\n",
salaries);
  printf("Address of the array distances is %x\n",
distances);
 }
```

Tip0510

```
#include <stdio.h>

void main(void)
 {
  int count[10];
  float salaries[5];
  long distances[10];

  printf("Address of the array count is %x &count is
%x\n",
    count, &count);
  printf("Address of the array salaries is %x &count is
%x\n",
    salaries, &salaries);
  printf("Address of the array distances is %x
&distances is %x\n",
    distances, &distances);
 }
```

Tip0511

```
#include <stdio.h>

void main(void)
 {
  int *iptr;     // Declare pointer variable
  int count = 1;

  iptr = &count;
  printf("Value of iptr %x Value of count %d Address of
count %x\n",
    iptr, count, &count);
 }
```

Tip0513

```
#include <stdio.h>

void main(void)
 {
  int counter = 10;
  int *iptr;        // Declare pointer value

  iptr = &counter;    // Assign the address
  printf("Addres in iptr %x Value at *iptr %d\n", iptr,
*iptr);

  *iptr = 25;       // Change the value in memory

  printf("Value of counter %d\n", counter);
 }
```

Tip0514

```
#include <stdio.h>

void swap_values(int *a, int *b)
 {
  int temp;

  temp = *a;  // Temporarily hold the value pointed to
by a
  *a = *b;    // Assign b's value to a
  *b = temp;  // Assign a's value to b
 }

void main(void)
 {
  int one = 1, two = 2;

  swap_values(&one, &two);

  printf("one contains %d two contains %d\n", one,
two);
 }
```

Tip 0516

```
#include <stdio.h>

void main(void)
 {
  int values[5] = {1, 2, 3, 4, 5};
  int counter;
  int *iptr;

  iptr = values;

  for (counter = 0; counter < 5; counter++)
   {
    printf("%d\n", *iptr);
    iptr++;
   }
 }
```

Tip0518

```
#include <stdio.h>

void show_string(char *string)
 {
  while (*string)
   putchar(*string++);
 }

void main(void)
 {
  show_string("Jamsa\'s C/C++ Programmer\'s Bible");
 }
```

Tip0520

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

char *string_uppercase(char *string)
 {
  char *starting_address;

  starting_address = string;

  while (*string)
   toupper(*string++);

  return(starting_address);
 }

void main(void)
 {
  char *title = "Jamsa\'s C/C++ Programmmer\'s Bible";
  char *string;

  string = string_uppercase(title);
  printf("%s\n", string);

  printf("%s\n", string_uppercase("Arrays and
Pointers"));
 }
```

Tip0523

```
#include <stdio.h>

void main(void)
 {
  char *weekdays[7] = {"Sunday", "Monday",
"Tuesday", "Wednesday",
                "Thursday", "Friday", "Saturday"};

  int i;

  for (i = 0; i < 7; i++)
   printf("weekdays[%d] contains %s\n", i,
weekdays[i]);
 }
```

Tip0525

```
#include <stdio.h>

void main(void)
 {
  char *workdays[] = {"Monday", "Tuesday",
"Wednesday",
                "Thursday", "Friday", "" };
  char **work_day;

  work_day = workdays;

  while (*work_day)
   printf("%s\n", *work_day++);
 }
```

Tip0529

```
#include <stdio.h>

int get_result(int a, int b, int (*compare)())
 {
  return(compare(a, b));  // Invoke the function passed
 }


int max(int a, int b)
 {
  printf("In max\n");
  return((a > b) ? a: b);
 }

int min(int a, int b)
 {
  printf("In min\n");
  return((a < b) ? a: b);
 }

void main(void)
 {
  int result;

  result = get_result(1, 2, &max);
  printf("Max of 1 and 2 is %d\n", result);

  result = get_result(1, 2, &min);
  printf("Min of 1 and 2 is %d\n", result);
 }
```

Tip0530

```
#include <stdio.h>

int what_is_the_value(int ***ptr)
 {
  return(***ptr);
 }

void main(void)
 {
  int *level_1, **level_2, ***level_3, value = 1001;

  level_1 = &value;
  level_2 = &level_1;
  level_3 = &level_2;

  printf("The value is %d\n",
what_is_the_value(level_3));
 }
```

Tip0538

```
#include <stdio.h>

struct Shape
{
 int type;
 int color;
 float radius;
 float area;
 float perimeter;
};

void show_structure(struct Shape shape)
 {
  printf("shape.type %d\n", shape.type);
  printf("shape.color %d\n", shape.color);
  printf("shape.radius %f shape.area %f
shape.perimeter %f\n",
    shape.radius, shape.area, shape.perimeter);
 }

void main(void)
 {
  struct Shape circle;

  circle.type = 0;
  circle.color = 1;
  circle.radius = 5.0;
  circle.area = 22.0 / 7.0 * circle.radius * circle.radius;
  circle.perimeter = 2.0 * 22.0 / 7.0 * circle.radius;

  show_structure(circle);
 }
```

Tip0539

```
#include <stdio.h>

struct Shape
{
 int type;
 int color;
 float radius;
 float area;
 float perimeter;
};

void change_structure(struct Shape *shape)
 {
  (*shape).type = 0;
  (*shape).color = 1;
  (*shape).radius = 5.0;
  (*shape).area = 22.0 / 7.0 * (*shape).radius *
(*shape).radius;
  (*shape).perimeter = 2.0 * 22.0 / 7.0 *
(*shape).radius;
 }

void main(void)
 {
  struct Shape circle;

  change_structure(&circle);

  printf("circle.type %d\n", circle.type);
  printf("circle.color %d\n", circle.color);
  printf("circle.radius %f circle.area %f circle.perimeter
%f\n",
    circle.radius, circle.area, circle.perimeter);
 }
```

Tip0541

```
#include <stdio.h>

struct Shape
{
 int type;
 int color;
 float radius;
 float area;
 float perimeter;
};

void change_structure(struct Shape *shape)
 {
  shape->type = 0;
  shape->color = 1;
  shape->radius = 5.0;
  shape->area = 22.0 / 7.0 * shape->radius * shape-
>radius;
  shape->perimeter = 2.0 * 22.0 / 7.0 * shape->radius;
 }

void main(void)
 {
  struct Shape circle;

  change_structure(&circle);

  printf("circle.type %d\n", circle.type);
  printf("circle.color %d\n", circle.color);
  printf("circle.radius %f circle.area %f circle.perimeter
%f\n",
    circle.radius, circle.area, circle.perimeter);
 }
```

Tip0544

```
#include <stdio.h>

void main(void)
 {
  struct Shape
           {
   int type;
   int color;
   float radius;
   float area;
   float perimeter;
  } circle = {0, 1, 5.0, 78.37, 31.42};

  printf("circle.type %d\n", circle.type);
  printf("circle.color %d\n", circle.color);
  printf("circle.radius %f circle.area %f circle.perimeter
%f\n",
    circle.radius, circle.area, circle.perimeter);
 }
```

Tip0547

```
#include <stdio.h>

void main(void)
 {
  struct Date
           {
   char month_name[64];
   int  month;
   int  day;
   int  year;
  } current_date = { "July", 7, 4, 1994 };

  int i;

  for (i = 0; current_date.month_name[i]; i++)
   putchar(current_date.month_name[i]);
 }
```