```
#include <stdio.h>
int main(){
  char ch;
  printf("enter a character: ");
  scanf("%c", &ch);
  if(ch \ge 'A' \&\& ch \le 'Z' \parallel ch \ge 'a' \&\& ch \le 'z')
     printf("%c is alphabet", ch);
  }
  else if(ch >= '0' && ch <= '9' ){
     printf("%c is digit", ch);
   }
  else{
     printf("%c is special symbol", ch);
  }
  return 0;
}
```

Output:

Test Case 1:

```
enter a character: #
# is special symbol
```

Test Case 2:

```
enter a character: 1
1 is digit
```

Test Case 3:

```
enter a character: a a is alphabet
```

```
Code:
```

equilateral

```
#include <stdio.h>
int is Valid Triangle (double side 1, double side 2, double side 3)
  if (side1 + side2 \le side3) return 0;
  if (side2 + side3 <= side1) return 0;
  if (side3 + side1 <= side2) return 0;
  return 1;
}
char* typeOfTriangle(double side1, double side2, double side3){
  if(side1 == side2 && side2 == side3) return "equilateral";
  if(side1 == side2 \parallel side2 == side3 \parallel side1 == side3) return "isosceles";
  return "scalene";
}
int main()
  double side1, side2, side3;
  printf("Enter all three sides of a triangle: ");
  scanf("%lf %lf %lf", &side1, &side2, &side3);
  if(isValidTriangle(side1, side2, side3)){
     printf("%s", typeOfTriangle(side1, side2, side3));
  else{
     printf("Invalid triangle");
  return 0;
Output:
Test Case 1:
Enter all three sides of a triangle: 1 2 3
Invalid triangle
Test Case 2:
 Enter all three sides of a triangle: 4 4 7
isosceles
Test Case 3:
 Enter all three sides of a triangle: 4 5 7
 scalene
Test Case 4:
 Enter all three sides of a triangle: 4 4 4
```

}

```
#include <stdio.h>
int main(){
  // taking the input
  int sub1Marks, sub2Marks, sub3Marks;
  printf("enter the masks of all three subjects: ");
  scanf("%d %d %d", &sub1Marks, &sub2Marks, &sub3Marks);
  //calculating the percentage
  const int TOTAL MARKS = 300;
  const long int SEM_FEE = 125000;
  int obtained = sub1Marks + sub2Marks + sub3Marks;
  float percentage = (float)obtained / TOTAL_MARKS * 100;
  int scholarship = 0;
  if (percentage < 50) scholarship = 0;
  else if(percentage < 61) scholarship = 5;
  else if(percentage < 75) scholarship = 20;
  else if(percentage < 85) scholarship = 30;
  else scholarship = 50;
  long int discount = SEM_FEE * scholarship / 100;
  long int payableAmount = SEM_FEE - discount;
  printf("Obtained Marks: %d\n", obtained);
  printf("Your Percentage %f%%\n", percentage);
  printf("Your Semester Fee is : %lld\n", SEM_FEE);
  printf("Your Scholarship is : %d%%\n", scholarship);
  printf("You Got Discount of : %lld%\n", discount);
  printf("Your Net Payable Ammount : %lld%\n", payableAmount);
  return 0;
```

Output:

Test Case 1:

```
enter the masks of all three subjects: 30 40 50
Obtained Marks: 120
Your Percentage 40.000000%
Your Semester Fee is: 125000
Your Scholarship is: 0%
You Got Discount of: 0
Your Net Payable Ammount: 125000
```

Test Case 2:

```
enter the masks of all three subjects: 50 60 70
Obtained Marks: 180
Your Percentage 60.000004%
Your Semester Fee is: 125000
Your Scholarship is: 5%
You Got Discount of: 6250
Your Net Payable Ammount: 118750
```

Test Case 3:

```
enter the masks of all three subjects: 60 70 80
Obtained Marks: 210
Your Percentage 70.000000%
Your Semester Fee is : 125000
Your Scholarship is : 20%
You Got Discount of : 25000
Your Net Payable Ammount : 100000
```

Test Case 4:

```
enter the masks of all three subjects: 80 90 95
Obtained Marks: 265
Your Percentage 88.333336%
Your Semester Fee is: 125000
Your Scholarship is: 50%
You Got Discount of: 62500
Your Net Payable Ammount: 62500
```

```
#include <stdio.h>
int isLeap(int year)
{
    if (year % 100 == 0 && year % 400 == 0) return 1;
    if (year % 100 != 0 && year % 4 == 0) return 1;
    return 0;
}

int main()
{
    int year;
    printf("Enter the year: ");
    scanf("%d", &year);
    printf(isLeap(year) ? "leap" : "not leap");
    return 0;
}
```

Output:

Test Case 1:

```
Enter the year: 2400 leap
```

Test Case 2:

```
Enter the year: 2100 not leap
```

```
#include <stdio.h>
int isPalindrome(int num)
  int rev = 0;
  int temp = num;
  while (temp)
     rev = rev * 10 + (temp % 10);
     temp = 10;
  return rev == num;
}
int main()
  int n;
  printf("Enter the n: ");
  scanf("%d", &n);
  int i, count;
  i = count = 1;
  while (count <= n)
     if (isPalindrome(i))
       printf("%3d %d\n", count++, i);
     i++;
  return 0;
}
```

Enter the n: 40	11 22	21 121	31 222
1 1	12 33	22 131	32 232
2 2	13 44	23 141	33 242
3 3	14 55	24 151	34 252
4 4	15 66	25 161	35 262
5 5	16 77	26 171	36 272
6 6 7 7	17 88	27 181	37 282
8 8	18 99	28 191	38 292
9 9	19 101	29 202	39 303
10 11	20 111	30 212	40 313

```
#include <stdio.h>
#include <math.h>
void convert(long long num, char type) {
  int mask, maskSize;
  int size = sizeof(num) * 8;
  const char *prefix;
  switch (type) {
     case 'b': (mask = 0b0001, maskSize = 1, prefix = "0b"); break;
     case 'o': (mask = 0b0111, maskSize = 3, prefix = "0"); break;
     case 'h': (mask = 0b1111, maskSize = 4, prefix = "0x"); break;
    default: printf("Invalid Type\n"); return;
  }
  printf("%s", prefix);
  int segments = ceil((float)size / maskSize) - 1;
  int val, leadingZeros = 1;
  for(int i = segments; i >= 0; --i){
     val = num >> (i * maskSize) & mask;
     if(val != 0) leadingZero = 0;
     if(val > 9){
       printf("%c", 'a' + val % 10);
     }
    else{
       printf("%d", val);
     }
  }
  printf("\n");
}
int main() {
  long long int decimalNum;
  printf("enter a decimal number: ");
  scanf("%lld", &decimalNum);
  convert(decimalNum, 'h');
  convert(decimalNum, 'o');
  convert(decimalNum, 'b');
  return 0;
}
```

Output:

enter a decimal number: 526023479

0x1f5a7b37 03726475467

0b11111010110100111101100110111

```
#include <stdio.h>
char* digitToString(int digit){
  switch (digit)
  {
     case 0: return "Zero";
     case 1: return "One";
     case 2: return "Two";
     case 3: return "Three";
     case 4: return "Four";
     case 5: return "Five";
     case 6: return "Six";
     case 7: return "Seven";
     case 8: return "Eight";
     case 9: return "Nine";
     default: return "";
  }
}
void print(int num){
  if(!num) return;
  print(num / 10);
  printf("%s ", digitToString(num % 10));
}
int main(){
  int num;
  printf("enter a number: ");
  scanf("%d", &num);
  print(num);
  return 0;
}
```

```
enter a number: 1234
One Two Three Four
```

```
#include <stdio.h>
#include <math.h>
double series1(int i, int fact){
  return pow(i, i)/fact;
double series2(int i, int fact){
  return pow(i, i-1)/fact;
double series3(int i, int fact){
  return (i & 1)? fact : -fact;
double accumulat(int n, double (*callback)(int, int)){
  double sum = 0;
  int fact = 1;
  for(int i = 1; i \le n; ++i){
     fact *= i;
     sum += callback(i, fact);
  }
  return sum;
}
int main(){
  int n;
  printf("enter the nth term: ");
  scanf("%d", &n);
  printf("Series 1: %lf\n", accumulat(n, series1));
  printf("Series 2: %lf\n", accumulat(n, series2));
  printf("Series 3: %lf\n", accumulat(n, series3));
  return 0;
}
```

```
enter the nth term: 10
Series 1: 4511.870241
Series 2: 491.729169
Series 3: -3301819.000000
```

```
#include <stdio.h>
int HCF(int num1, int num2){
  int rem;
  do
    rem = num1 % num2;
    num1 = num2;
    num2 = rem;
  \} while (num2 != 0);
  return num1;
}
int LCM(int num1, int num2){
  return (num1 * num2) / HCF(num1, num2);
}
int main(){
  int num1, num2;
  printf("Enter two numbers: ");
  scanf("%d %d", &num1, &num2);
  printf("LCM: %d\n", LCM(num1, num2));
  printf("HCF: %d\n", HCF(num1, num2));
}
```

```
Enter two numbers: 12 18
HCF: 6
LCM: 36
```

```
#include <stdio.h>
int main(){
  int n;
  printf("Enter the nth Term: ");
  scanf("%d", &n);
  // ====== pattern 1
  for(int i = 1; i <= n; ++i){
     for(int j = 1; j \le n - i; ++j){
        printf(" ");
     for(int j = 1; j <= i; ++j){
        printf("%d", j);
     for(int j = i - 1; j >= 1; --j){
        printf("%d", j);
     printf("\n");
  for(int i = 2; i <= n; ++i){
     for(int j = 1; j < i; ++j){
        printf(" ");
     for(int j = 1; j \le n + 1 - i; ++j){
        printf("%d", j);
     for(int j = n - i; j >= 1; --j){
        printf("%d ", j);
     printf("\n");
   }
  // ===== pattern 2
  printf("\langle n \rangle n");
  for(int i = 1; i <= n; ++i){
     for(int j = 1; j \le n - i; ++j){
        printf(" ");
     for(int j = n; j > n - i; j - -){
        printf("%c ", j + 'A' - 1);
     for(int j = n+2-i; j <= n; ++j){
        printf("%c", j + 'A' - 1);
     printf("\n");
  for(int i = 2; i <= n; ++i){
```

```
for(int j = 1; j < i; ++j) \{ \\ printf(" "); \\ \} \\ for(int j = n; j >= i; j--) \{ \\ printf("%c ", j + 'A' - 1); \\ \} \\ for(int j = i + 1; j <= n; ++j) \{ \\ printf("%c ", j + 'A' - 1); \\ \} \\ printf("\n"); \\ \} \\ return 0; \\ \}
```

```
Enter the nth Term: 5
           1
        1 2 1
     1 2 3 2 1
  1 2 3 4 3 2 1
1 2 3 4 5 4 3 2 1
  1 2 3 4 3 2 1
     1 2 3 2 1
        1 2 1
           1
           Е
        E D E
     EDCDE
  EDCBCDE
\mathsf{E}\;\mathsf{D}\;\mathsf{C}\;\mathsf{B}\;\mathsf{A}\;\mathsf{B}\;\mathsf{C}\;\mathsf{D}\;\mathsf{E}
  EDCBCDE
     EDCDE
        E D E
           Е
```

```
#include <stdio.h>
struct Time{
  int hour;
  int minut;
  int second;
};
void input(struct Time *time){
  printf("enter time (hh mm ss) ");
  scanf("%d %d %d", &time->hour, &time->minut, &time->second);
}
void output(struct Time *time){
  printf("%02d:%02d:%02d\n", time->hour, time->minut, time->second);
}
struct Time sum(struct Time *t1, struct Time *t2){
  struct Time result;
  result.second = (t1->second + t2->second);
  result.minut = (t1->minut + t2->minut) + (result.second / 60);
  result.hour = (t1->hour + t2->hour) + (result.minut / 60);
  result.second %= 60;
  result.minut %= 60;
  result.hour %= 60;
  return result;
}
int main(){
  struct Time t1, t2, t3;
  input(&t1);
  input(&t2);
  t3 = sum(\&t1, \&t2);
  output(&t1);
  output(&t2);
  output(&t3);
}
```

```
enter time (hh mm ss) 12 45 30
enter time (hh mm ss) 03 20 45
12:45:30
03:20:45
16:06:15
```