①

```
#include <iostream>
using namespace std;
int main (){
 char arr[9] = "worksheet";
 for (int i=0; i<9; i++){
    for (int j=0; j< 9-i ; j++){
    if ( arr[j] > arr[j+1]){
        temp = arr[j] ;
        arr[j] = arr[j+1];
        arr[j+1] = arr[j] temp;
      }
  for (int j=0 ; j<n ; j++){
    cout<< arr[j] <<endl;
    }
  return 0;
   }
```

②

```
#include <iostream>
using namespace std;
int main (){
 int m, n, a, b;
 int arr[m][n];
 //order of first matrix  m x n
 // order of second matrix  a x b
   cout<<order of first matrix<<endl;
  cin >> m >>endl;
   cin >> n >>endl;
 cout << order of second matrix <<endl;
   cin >> a >>endl;
   cin >> b >> endl;
 if ( n==a){
```

```
cout << 1st matrix input << endl;
// taking matrix inputs
for (int i=0; i<m; i++){
    for (int j=0; j<n; j++){
    cin >> arr1[i][j];
}
for (i
  cout << 2nd matrix input << endl;
for (int i=0; i<a; i++){
  for (int j=0; j<b; j++){
    cin >> arr2[i][j];
  }
// printing the 2 matrices
for (int i=0; i<m; i++){
  for (int j=0; j<n; j++){
    cout << arr1[i][j];
  }
    cout << 2nd matrix << endl;
  for (int i=0; i<a; i++){
    for (int j=0; j<b; j++){
      cout << arr2[i][j];
    }
if (n == a){
  for (int i=0; i<m; i++){
    for (int j=0; j<n; j++){
  sum=0;
    for (int k=0; j<n; k++){
      sum = sum + arr1[i][k] * arr2[k][j];
    }   c[i][j] = sum;
  }
  }
  cout << }
    for (int i=0; i<m; i++){
      for (int j=0; j<n; j++){
        cout << c[i][j] << endl;
        cout << " " << endl;
      }
    n;}
```

③

```
#include <iostream>
using namespace std;
int main() {
int arr[3][3] = {};
  for (int i ; i <3 ; i++) {
    for (int j ; j <3 ; j++) {
     cin >> arr[i][j];
    }
  }

  for (int i; i<3 ; i++) {
  for (int j; j<3 ; j++) {
    cout >> arr[i][j];
  }
  }

  for (int i ; i<3 ; i++) {
    for (int j ; j <3 ; j++) {
      if (arr[i][j] == 0) {
        for (int k ; k <3 ; k++) {
          arr[i][k] = 0 ;
          arr[k][j] = 0;
        }
      }
    }
```

```
    cout << "the new matrix is" << endl;
// printing the new matrix
for(int i=0; i<3; i++) {
    for(int j=0; j<3; j++) {
    cout << arr[j][i];
    }
    }
return 0;
}
```

```
}
```

④

```
#include<iostream>
using namespace std;
int main() {
int arr[10] = {1,
for(int i=0; i<10; i++){
cin >> arr[i];
}
for (int i=0; i<10; i++) {
if (arr[i] < arr[i+1]) {
continue;
}
else {
if (arr[i+2] < arr[i+1])
count == 1;
else {count = 0; break;}
}
if (count == 1) continue;
if else (count == 0) break; {
cout "Not a hill number" << endl;
break;
}
}
if (count == 1) { cout << Is a hill number <<
```

⑥
```
#include <iostream>
using namespace std
int main () {
 int num1 , num2;
 cin>> num1;
 cin>> num2;
 int sum=0;
 for (int i=2; i<num1 ; i++) {
   if (num1 % i ==0) {
      sum= sum+ i;
   }
  }
 if (sum == num2) {
   cout <<          ;
  }
 else  cout<<          ;
 return 0;
 }
```

⑥
```
#include <iostream>
using namespace std
int bubblesort () {
 for (int i = 0; i<n-1; i++) {
   for (int j = 0; j<n-i-1 ; j++) {
     if ( arr[j+1] < arr[j]) {
       temp = arr[j];
       arr[j] = arr[j+1];
       arr[j+1] = temp;
     }
   }
 }
```

```
for (int i=0 ; i < n ; i++) {
  cout << arr[i] << endl;
}
  return arr;
}
int selectionsort () {
for(int i=0; i<n-1 ; i++){
  for (int j=i+1; j<n; j++) {
   if (arr[i] > arr[j]) {
     temp = arr[i];
     arr[i] = arr[j];
     arr[j] = temp;
     }
  for (int i=0; i<n; i++){
     cout << arr[i] << endl;
    }
    return arr;
    }
int binarysearch (int num) {
  low = 0
  high = n-1
  do {
  mid = (low + high) / 2;
  if (mid < num) {
       low = mid+1;
     }
   else high = mid-1
   } while (num != mid && low <= high);
    mid = (low+high)/2;
   if (num == mid) {
    cout (" successful search. element found at:
                 " << mid << endl;
```

```
cout <<"enter first matrix"<< endl;
for (int i=0; i<m; i++){
  for (int j=0; j<n; j++){
    cin>> A[i][j];
      }
      }
cout << "enter second matrix "<< endl;
for (int i=0; i<a; i++){
for (int j=0; j<m; j++){
    cin>> B[i][j];
    }
    }
multiplication (A, B);
  transpose (C);
   transpose (A);
  transpose (B);
   if (C == B*A){
     cout <<" (A.B)' = B'.A' "<< endl;
   }
  return 0;
   }
```

URDAN EDGE

(8)

```
#include<iostream>
using namespace std
int multiplication (int arr1 , int arr2){
for( int i=0; i<n; i++){
  for (int j = 0; j<m; j++){
  for (int k =0; k<m; k++){
    arr [i][j] = arr [i][k] * arr [k][j]
        sum = sum + arr [i][j];
        c [i][j] = sum;
        }
        }
       sum =0;
       }
   for (int i=0; i<n ; i++) {
     for (int j=0; j<b; j++) {
       cout << c[i][j] << endl;
       }
       }
     return ;
     }
int transpose (int arr) {
   for (int i=0 ; i<n; i++) {
    for (int j=0; j<m; j++){
     cout << arr[j][i] << endl;
     }
     }
     }
int main () {
  int m=3, n=3, a=3, b=3;
  int A[m][n], B[a][b], c[m][b];
  int sum=0;
```

```
return mid;
  }
int main () {
  int arr [5] = { 2, 4, 10, 5, 64};
  char sort;
  cout << " enter s or b." <<endl;
  cin >> sort ;
  if (sort == s) {
       selectionsort (arr); }
if else (sort == b) {
       bubblesort ( arr);
     }
  // Binary Search
  cin >> num >>endl;
  cout<< " the no. to be searched:" <<num<<end
   binarysearch (int num);
  return 0;
  }
```

(7)

```
#include <iostream>
#include <stdio.h>
using namespace std;
int main ()
{
 char str1[100], str2[100],
 char * str1Ptr, *str2Ptr;
 cout << "Enter the string: ";
 gets (str1);
 str1 = & str1[0];
 str2 = & str2[0];
 while (* str1Ptr)
 {
   * str2Ptr = * str1Ptr
     str1Ptr++;
     str2Ptr++;
  }
  * str1 = '\0';
  cout << "\n Enter string: " << str1;
  cout << "\n copied string: " << str2;
   cout << endl;
   return 0;
  }
```

decimal → hexadecimal

5(b)

```
#include <iostream>
using namespace std;
int main()
{
int num, rem, i=0;
char hexanum[100];
cout << "Enter the decimal number: ";
cin >> decimal num;
while (num!=0)
{
rem = num%16;
if (rem<10)
  rem = rem+48;
else
  rem = rem+55;
hexanum[i] = rem;
i++;
 num = num/16;
}
cout << " Hexadecimal representation of the number is: ";
for (int i=0; i<=0; i--)
   cout << hexanum[i];<<endl;
cout.
return 0;
}
```

⑤ a)
```
#include <iostream>
using namespace std
int rev(int x){
  int i=0;
  while (num>0){
    rem[i] = num % x;
    num = num / x;
    i++;
  }
  for(int j=i-1; j>=0; j--){
      cout << rem[j] << endl;
    }
  return rem;
  }
cout <<
  int main(){
  int rem[], num, y;
  cin >> num; y=num;
  cout << Binomial "decimal number is: " << endl;
    cin >> num;
  cout << " Binomial representation" << endl;
   rev(2);
num=y  cout << " Hexadecimal representation" << endl;
   cout << rev(x);
    return 0;
    }
  cout << " octal representation" << endl;
   cout << rev(8) << endl;
   return 0;
   }
```

URBAN EDGE

(9)

```
#include <iostream>  - using namespace std
void sorting(length(string[][])
{
  for (int i=0; i< m; i++){
  if ( length(string[i][]) > length(string[i+1][]))
    {
      temp = string[i][];
      string[i][] = string[i+1][];
      string[i+1][] = temp;
    }
  cout << string[i][] << endl;
  }
}
void sorting(alphabetically( str[]);
  int n = length(str[]);
  for (int j=0; j<n-1; j++){
    if ( str[j] > str[j+1]) {
      temp = str[j];
      str[j] = str[j+1]
      str[j+1] = temp;
    }
  cout << str[j] << endl;
  }
int main()
{
  char string[3][30];
  int temp;
  for (int i=0; i<3; i++){
    for (int j=0; j<30; j++){
      cin >> string[i][j];
      cout << string[i][j]
    }
  }
}
```

URBAN EDGE

(10)

```
#include <iostream>
using namespace std

int fibonacci (int num){
  if (num == 0){
    return 0;
    }
if else (num == 1) {
    return 1;
    }
else {return (fibonacci (num-1) + fibonacci (num-2));
  }
  }
  int main (){
  int num;
  cin>>num;
  for (int i=0; i < num; i++) {
    cout << fibonacci (i) <<endl;
    }
  return 0;
    }
```

```
sortinglength ( string[30][30] );
    for (int i=0; i<3; i++) {
        sortingalphabetically (string[i]);
    return 0;
    }
```

}

(11)

```
#include <iostream>
using namespace std
int binarysearch (int arr[], int low, int high, int num)
    if (high >= 1){
        mid = (low + high)/2;
    if (arr[mid] == num) {
        return mid;
        }
```

```
else {
  if (arr[mid] > num){
    return binarysearch(arr[], low, mid-1, num);
    }
  else {
    return binarysearch(arr[], mid+1, high, num);
    }
  return 2; //if element not found
  }
int main() {
  int num, low, high, mid; index;
  int arr[] = {60, 61, 82, 92, 108};
  int high = size of(arr) / size of(arr[0]);
  low = 0;
  int index = binarysearch(arr, 0, high-1, num);
    if (index == 2) {
      cout << "searcher number not found" << endl;
      else
      cout << "number succesfully found at" << index;
      return 0;
      }
```

(12)

```
#include <iostream>
using namespace std
int factorial (int num);
 int f=1;
 if (num==0) {
    return 0;
      }
   else {
return f= f* factorial (num-1);
    }
 int main () {
   int number, fact;
   cin >> number;
   fact = factorial (number);
   cout << fact <<endl;
   return 0;
   }
```

(14)

```
#include <iostream>
using namespace std
int main()
{
 int num1, num2, arr1[100], arr2[100], gcd = -100;
  cin >> num1;
  cin >> num2;
 for (int i=1; i< num1 ; i++) {
     for (int j=0; j<100; j++) {
   if ( num1 % i == 0) {
       arr1[j] = i;
       }
       }
       }
 for (int i=1; i<num2; i++) {
   for (int j=0; j<100; j++) {
   if (num2 % i == 0) {
       arr2[j]=i;
       }
       }
       }
 for (int i=0; i <100; i++) {
   for (int j=0; j<100 ; j++) {
    if (arr1[i] == arr2[j]) {
          if (gcd < arr1[i]) {
            gcd = arr1[i];
            }
            }
            }
  cout << gcd << endl;
   return 0;
   }
```

(13) Using Recursion

```
#include <iostream>
using namespace std
int sumofdigits(int num)
{int sum;
if (num==0)
return 0;
else {sum= num%10+sumofdigits(num/10);
return sum;}
}
int main(){
int number;
cin>>number;
cout<<"sum of digits is: "<<endl;
cout<<sumofdigits(number)<<endl;
return 0;
}
```

(13) without using recursion

```
#include<iostream>
using namespace std
int main(){
int num, sum=0;rem;
cin>>num;
do{
 rem = num%10;
 num = num/10;
 sum = sum + rem;
}while(num>0);
cout<<sum<<endl;
return 0;
}
```

(15)

```
#include <iostream>
using namespace std;
void bubblesort (int arr[], int n){
  if (n==1) return;
 for (int i=0; i<=n-2; i++){
    if (arr[i]> arr[i+1]){
     int temp= arr[i+1];
      arr [i+1]= arr[i];
        arr[i]= temp;
    }
  }
   bubblesort (arr, n-1);
  }
int main ()
 {
 int arr[]= {1, 48, 101, 26, 2, 8};
  int n= sizeof(arr)/sizeof(arr[0]);
  cout <<"Sorted list is: "<<endl;
  for ( int i=0; i<n; i++)
   {
    cout << arr[i]<<" ";
    }
```

```
cout <<endl;
bubblesort(arr,n);
cout << "Sorted list is : " << "\n";
for(int i=0;i<n;i++){
    cout <<arr[i]<<" ";
}
cout << "\n";
return 0;
}
```