```cpp
 /*"1"-----------------------------------------
C++ function call by reference :
*/

#include <iostream>
#include <string>
using namespace std;
void swapvariable(int &x, int &y) {
    int temp;
    temp = x;
    x = y;
    y = temp;

    return;
}


void swapvariable(int &x, int &y);

int main () {
    int a = 100;
    int b = 200;

    cout << "Before swap, value of a :" << a << endl;
    cout << "Before swap, value of b :" << b << endl;

    swapvariable(a, b);

    cout << "After swap, value of a :" << a << endl;
    cout << "After swap, value of b :" << b << endl;

    return 0;
}
```

```cpp
/*"2"------------------------------------------------------
--------------------------
function to print and insert array */

#include <iostream>
using namespace std;
 int arr[20];
void inp(int a)
{
    for (int i=0;i<a;i++)
    {
    cout<<"enter number "<<endl;
    cin>>arr[i];
    }
}
void prin(  int a)
{
 for (int i=0;i<a;i++)
    {
    cout<<arr[i]<<endl;
    }
}

int main() {
int n;
        cout<<"array size"<<endl;
cin>>n;

inp(n);
prin(n);
```

```cpp
    return 0;
}

/*"3"----------------------------------------
print array using function  and how to return array */

#include <iostream>
using namespace std;

int main ()
{
   int numbers[5];
   int * p;
   p = numbers;   *p = 10;
   *p = 20;

   p = &numbers[2];   *p = 30;
   p = numbers + 3;   *p = 40;
   p = numbers;   *(p+4) = 50;
   for (int n=0; n<5; n++)
     cout << numbers[n] << ", ";
   return 0;
}

/*"example 2"----------------------------------------
----------------------------------
access value using pointer */

#include <iostream>
using namespace std;

int main ()
```

```cpp
{
  int firstvalue, secondvalue;
  int * mypointer;

  mypointer = &firstvalue;
  *mypointer = 10;
  mypointer = &secondvalue;
  *mypointer = 20;
  cout << "firstvalue is " << firstvalue << '\n';
  cout << "secondvalue is " << secondvalue << '\n';
  return 0;
}

/*--------------------------------------------
print array using function  and how to return array*/

#include<iostream>
using namespace std;
 int a[5];
int* demo()
{
    for(int i = 0; i<5; i++)
    {
    cin>>   a[i];
    }

    return a;
}

int main()
{
    int* ptr;
    int i;
```

```cpp
    ptr = demo();
    cout<<"Array is: ";
    for(i=0 ; i<5; i++)
        cout<<ptr[i]<<"\t";

    return 0;
}


/*"5"-------------------------------------------------------
----------------------

print array using function  and how to return
array  :secondly */

#include <iostream>
#include<array>
using namespace std;

array<int,5> func()
{
    std::array<int,5>

    for(int i=0;i<5;i++)
    {
        f_array[i] = i;
    }

    return f_array;
}

int main()
{
```

```cpp
    std::array<int,5> arr;

    arr=func();

    cout<<"The Array is : ";
    for(int i=0;i<5;i++)
    {
        cout<<arr[i]<<"\t";
    }

    return 0;
}

/*some method of array with example :size:*/

#include <array>
#include <iostream>

int main()
{
    std::array<int, 4> nums{1, 3, 5, 7};

    std::cout << "nums contains " << nums.size() << "
elements.\n";
}

/*"6":emptyand at to set and return spacific value :*/
#include <array>
#include <iostream>

int main()
{
    array<int, 4> numbers{3, 1, 4, 1};
```

```cpp
    array<int, 0> no_numbers;

    scout << boolalpha;
    cout << "numbers.empty(): " << numbers.empty() <<
'\n';
    cout << "no_numbers.empty(): " << no_numbers.empty()
<< '\n';
    array<int, 6> data = {1, 2, 4, 5, 5, 6};

    data.at(1) = 88;

    cout << "Element at index 2 has value " << data.at(2)
<< '\n';

    scout << "data size = " << data.size() << '\n';
}

/*"7":front and back:*/

#include <array>
#include <iostream>

int main()
{
    std::array<char, 6> letters{'a', 'b', 'c', 'd', 'e',
'f'};

    if (!letters.empty())
        std::cout << "The first character is '" <<
letters.front() << "'.\n";
        std::cout << "The last character is '" <<
letters.back() << "'.\n";
```

```
}
```