EXPERIMENT NUMBER: 5

Objective: Write a program in C to perform the Power Set operation on a set.
Definition : Power Set Power set P(S) of a set S is the set of all subsets of S. For example S = {a, b, c} then P(s) = {{}, {a}, {b}, {c}, {a,b}, {a, c}, {b, c}, {a, b, c}}.

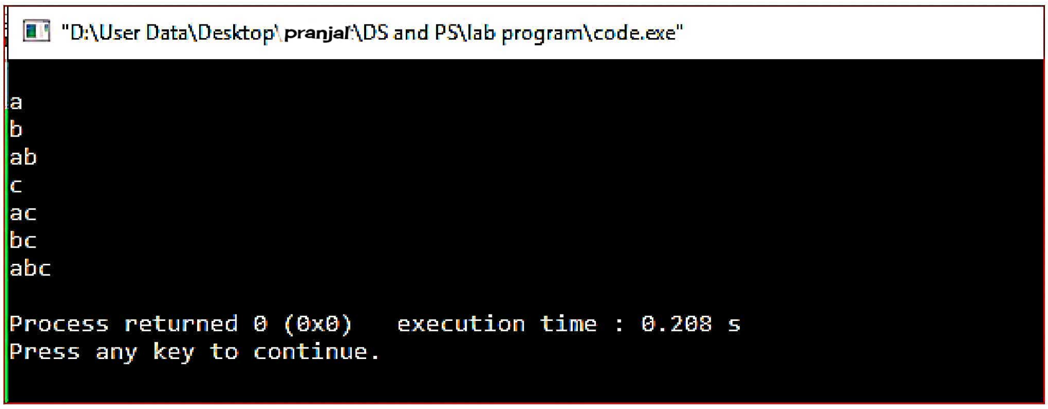
If S has n elements in it then P(s) will have 2^n elements

```
Input: Set[], set_size
1. Get the size of power set
   powet_set_size = pow(2, set_size)
2  Loop for counter from 0 to pow_set_size
    (a) Loop for i = 0 to set_size
        (i) If ith bit in counter is set
            Print ith element from set for this subset
    (b) Print separator for subsets i.e., newline
```

CODE:

```
#include <stdio.h>
#include <math.h>
 void printPowerSet(char *set, int set_size){
   unsigned int pow_set_size = pow(2, set_size);
   int counter, j;
   for(counter = 0; counter < pow_set_size; counter++){
     for(j = 0; j < set_size; j++){
        if(counter & (1<<j)) printf("%c", set[j]);
      }
      printf("\n");
   }
}
int main()
{
   char set[] = {'a','b','c'};
   printPowerSet(set, 3);
   return 0;
}
```

OUTPUT:



"D:\User Data\Desktop\ pranjal\DS and PS\lab program\code.exe"

```
a
b
ab
c
ac
bc
abc

Process returned 0 (0x0)   execution time : 0.208 s
Press any key to continue.
```

:

EXPERIMENT NUMBER: 7

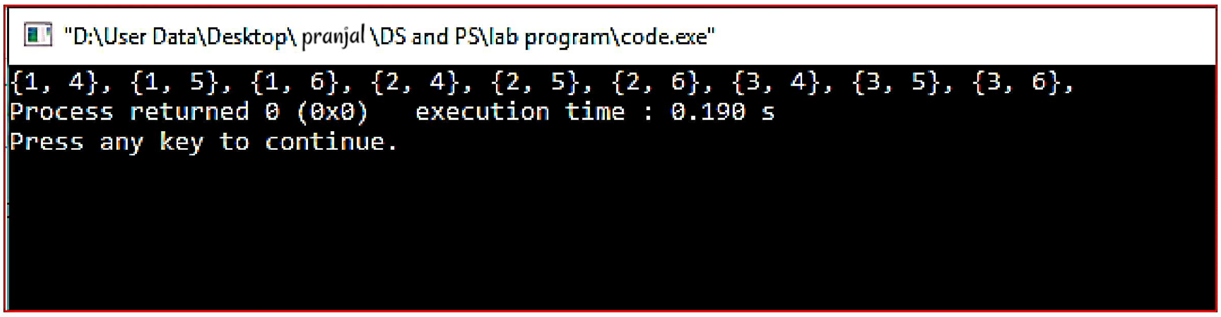
Objective: Write a program in C to perform the Power Set operation on a set.
Definition : Let A = {a, b, c} and B = {d, e, f} The Cartesian product of two sets is A x B = {a, d}, {a, e}, {a, f}, {b, d}, {b, e}, {b, f}, {c, d}, {c, e}, {c, f}} A has 3 elements and B also has 3 elements. The Cartesian Product has 3 x 3 = 9 elements.
In general, if there are m elements in set A and n elements in B, the number of elements in the Cartesian Product is **m x n**

CODE:

```
#include<stdio.h>
void CartesianProduct(int arr1[], int arr2[], int n, int n1){
   for (int i = 0; i < n; i++)
      for (int j = 0; j < n1; j++)
         printf("{%d, %d}, ", arr1[i], arr2[j]);
}

int main(){
   int arr1[] = { 1, 2, 3 }; // first set
   int arr2[] = { 4, 5, 6 }; // second set
   int n1 = sizeof(arr1) / sizeof(arr1[0]);
   int n2 = sizeof(arr2) / sizeof(arr2[0]);
   CartesianProduct(arr1, arr2, n1, n2);
   return 0;
}
```

OUTPUT:


"D:\User Data\Desktop\ pranjal \DS and PS\lab program\code.exe"

```
{1, 4}, {1, 5}, {1, 6}, {2, 4}, {2, 5}, {2, 6}, {3, 4}, {3, 5}, {3, 6},
Process returned 0 (0x0)   execution time : 0.190 s
Press any key to continue.
```