Name: Gadhiya Nimish Roll no. 25bcl094

Section - A

1. Accept 5 values and print them

```
#include <stdio.h>
int main(){
int arr[5], i;
printf("Enter 5 value: ");
for(i=0; i<5; i++) {
scanf("%d", &arr[i]); }
printf("the values are: ");
for(i=0; i<5; i++) {
printf("%d", arr[i]);
}
}
```

2. Accept 10 values and print 4th, 7th and 9th values.

```
#include<stdio.h>
int main(){
int value[10];
int i; printf("Enter 10 integer values:
for(i=0; i<10; i++){
printf("enter value %d:", i+1);
scanf("%d", &values[i]); }
printf("\n printing specific values:\n");
printf("4th value: %d\n", values[3]);
printf("7th value: %d\n", values[6]);
printf("9th value: %d\n", values[8]);
}
```

3. Accept 5 values and sort ascending / descending

```
#include <stdio.h>
    int main () {
      int arr[5], i, j, temp;
    printf(" Enter 5 values: ");
    for(i=0; i<5; i++) {
    scanf (" %d", &arr[i]);
    for(i=0; i<5; i++) {
     for(j=i+1; j<5; j++) {
      if(arr[i] > arr[j]) {
        temp = arr[i]; arr[i] = arr[j] = temp;
        }
      }
    }
    printf(" Ascending: ");
    for(i=0; i<5; i++) {
     printf ("%d", arr[i]);
     for(i=4; i>=0; i--) {
      printf ("%d", arr[i]);
      }
    }
```

Date:
Page No.:

4. Print minimum notes required

```
#include <stdio.h>
    int main () {
    int amount, notes[ ] = {500, 200, 100, 50,
                    20, 10, 5, 2, 1}, count, i;
    Printf (" Enter amount : " );
    scanf ( "%d" , &amount );
    for( i=0; i<9; i++ ) {
        Count = amount / notes [i];
    if(count) {
    Printf ( "%d x %d\n" , count, notes[i]);
    amount %= notes [i];
}
    }
}
```

5 add two 2 arrays.

```
#include <stdio.h>
int main ()
#include #define ROWS 3
#define COLS 3
int main () {
int array 1 [ROWS][COLS] = {
{1,2,3}, {4,5,6}, {7,8,9}
};

int array 2 [ROWS][COLS] = {
{9,8,7}, {6,5,4}, {3,2,1}
};
int sum array [ROWS][COLS];
for (int i=0; i<ROWS; i++) {
for (int j=0; j<COLS; j++) {
sum array [i][j] = array 1 [i][j] + array 2 [i][j];
}
}
printf ("sum of the two 2D arrays: ");
for (int i=0; i<ROWS; i++) {
for (int j=0; j<COLS; j++) {
printf ("%d", sum array [i][j]);
}
printf ("\n");
}
}
```

6. multiply two 2D array

```
#include <stdio.h>
#define R1 3
#define C1 2
#define R2 2
#define C2 3
int main () {
    int matrix1 [R1][C1];
    int matrix2 [R2][C2];
    int Result matrix [R1][C2];
    int i, j, k;
ppintf(" enter elements for first matrix
                    (%d x %d): n",
                    R1, C1 );
for(i=0; i<R1; i++){
for(j=0; j<C1; j++){
ppintf(" enterelemend matrix, [%d][%d]: ", i,j);
scanf ("%d", &matrix1 [i][5];
    }
    }

    ppintf(" enter elements for second matrix (%d x %d)",
                    R2, C2 );
for(i=0; i<R2; i++){
    for (j=0; j<C2; j++){
ppintf(" enter elemend matrix2 [%d][%d]: ", i, j );
    scanf ("%d", &matrix2 [i][j]);
    }
```

```
for(i=0; i<R1; i++) {
    for(j=0; j<C2; j++) {
        Result matrix[i][j]=0;
    }
}
for(i=0; i<R1; i++) {
    for(j=0; j<C2; j++) {
        for(k=0; k<C1; k++) {
            Result matrix[i][j]+=matrix[i][k]*matrix2[k][j];
        }
    }
}
printf("Result matrix multiplication
    (%dx%d):\n", R1, C2);
for(i=0; i<R1; i++) {
    for(j=0; j<C2; j++) {
        printf("%d ", Result matrix[i][j]);
    }
    printf("\n");
}
```

7. Obtain transpose of a 4x4 matrix

```
#include <stdio.h>
int main () {
int matrix[4][4];
int transpose[4][4];
int i, j;
Printf("enter element of 4x4 matrix: \n");
for (i=0; i<4; i++) {
  for (j=0; j<4 ; j++) {
   Printf("enter element matrix[%d][%d]: ",i,j);
scanf("%d", &matrix [i][j] );
   }
}
for(i=0; i<4 ; i++) {
for (j=0; j<4 ; j++) {
  transpose[i][j] = matrix[i][j];
   }
}
Printf ("\n Transposed matrix : \n");
  for (i=0; i<4; i++) {
   for (j=0; j<4; j++) {
Printf ("%d", matrix [i][j]);
}
 Printf ("\n");
}
```

```
printf("\n Transposed matrix: \n");
  for (j=0; j<4; j++){
    for(j=0; j<4; j++){
      printf("%d", transpose[i][j]);
      }
      printf("\n");
    }
  }
```

8 copy one array of 5 another of 10

```
#include <stdio.h>
int main() {
 int a[5], b[10]={20}, i;
 printf("enter 5 values: ");
 for(i=0; i<5; i++) {
 scanf("%d", &a[i]);
 for(i=0; i<5; i++) {
 b[i*2]=a[i];
 printf("copied array: ");
 for(i=0; i<10; i++) {
 printf("%d", b[i]);
  }
 }
 }
 return 0;
}
```

9. Reverse an array 5 elements

```
#Include <stdio.h>
int main () {
int arr[5], i;
printf(" enter 5 value: ");
for(i=0; i<5 ; i++) {
Scanf ("%.d", &arr [i];
}
printf ("Reversed:  ");
for(i=4; i>=0, i--) {
  Printf("%d", arr[i]);
    }
```

10. find freqeency of number in array

```
#include <stdio.h>
  int main () {
  int arr[10], freq[10]=[0], i, j, count;
printf("enter 10 values: ");
for(i=0; i<10; i++) {
  Scanf("%.d", &arr[i]);
  for(i=0; i<10; i++){
  count = 1
    if (freq[i]) continue;
 for (j=i+1; j<10; j++) {
if (arr[i] == arr[j]) {
count++; freq(j)=1; }
}
}
```

```
printf("%d occurs %d time", appli, count);
    }
    return 0;
    }
```

11. shift all number by n positions

```
#include <stdio.h>
#include<string.h>
Void shift array(int arr[] int size, int n, char
                                direction ) {
    int tempArr [size] ;
memset(tempArr, sizof(temp Arr);

if (direction == 'R' || direction == 'r') {
  for(int i=0 ; i<size ; i++) {
     if(i-n >= 0 ) {
          tempArr[i] = arr [i-n];
     }
  }
} else if (direction == 'L' || direction == ',' ) {
for( int i=0; i<size; i++ ) {
    if(i+n<size) {
     temp [i] = arr [i+n] ;
    }
}
}
```

```
    else {
        printf("Invalid direction use 'l' for left or 'r'
                for right \n");
    }
    for(int i=0; i<size; i++) {
        arr[i] = temparr[i];
    }
}
int main() {
    int arr[10] = {1,2,3,4,5,6,7,8,9,10};
    int size = 10;
    int n=3;
    char direction = 'R';
    printf("original array: ");
    for(int i=0; i<size; i++) {
        printf("%d", arr[i]);
    }
printf("\n");
shiftArray(arr, size, n, direction);
printf("Array after %d position %s shift: ", n,
(direction == 'r' || direction == 'R') ?
        "right" : "left");
}
printf("original array: ");
for(int i=0; i<size; i++) {
    printf("%d", arr[i]);
}
```

```
printf("\n")
shiftArray(array, size, n, direction);
printf("array after shifting %d position %c\n",
"right" : "left");
    for(int i=0; i<size; i++){
      printf("%d", array[i]);
    }
    printf("\n");
}
```

12 Insert number at beginning

```
#include <stdio.h>
int main(){
    int arr[10], n, i;
printf("enter 9 number: ");
for(i=1; i<10; i++){
scanf("%d", &arr[i]);
}
printf("enter a new number: ");
scanf("%d", &n);
    arr[0] = n
    printf("array: ");
for(i=0; i<10; i++){
printf("%d", arr[i]);
}
return 0;
}
```

13. Insert a new number at a particular position

```
#include<stdio.h>
int main() {
int appy[100], position, c, n, value;
printf("enter number of elements in array");
scanf("%d", &n);
printf("enter %d elements \n", n);
for ( c=0; c<n; c++) {
scanf("%d", &appay[c]);
printf("enter the loction where you with to
insert an element \n");
scanf("%d", &position);
printf("enter the value to insert \n");
scanf("%d", &value);
for (c=n-1; c>=position-1; c--) {
appay[c+1] = appay[c];
appay[position-1] = value;
printf("Resultant array is \n");
for (c=0; c<=n; c++){
printf("%d", appay[c];
}
}
```

14. Insert a new number at last no of array

```
#include <stdio.h>
    int main () {
int arr[100] = {1,2,3,4,5};
int size=5;
int new, number=6;
if (size<100) {
  arr[size] = new - number;
    size ++;
printf ("Array after insertion: ");
  for(int i=0; i<size; i++) {
   printf("%d", arr[i]);
  }
  printf ("\n");
}
else {
  printf ("Array is full cannot insert new
          number \n");
}
}
```

S | M | T | W | T | F | S

Date:

Page No.:

BOOK BUZZ

15. delet a value from the first position of an array

```
#include <stdio.h>
int main () {
int arr [100];
    int size, i;
printf("enter the size of the array: ");
    scanf("%d", &size);
printf("enter the elements of the array: ");
    for (i=0; i<size; i++) {
        scanf("%d", &arr[i]);
    }
        printf("\n");
} else {
    printf("Array is full cannot insert new
            number \n");
    }
}
```

16 Delet a value from a particular position in an array

```
#include <stdio.h>
int main () {
    int arr[100];
    int size , i, pos;
printf (" enter the size of the array: ");
    scanf (" %d", &size);
    printf("enter %d elements in the array: ");
    for (i=0; i<size; i++) {
        scanf ("%d", &arr[i]);
    }
    printf(" enter the position of the element to delete:");
scanf("%d", &pos);
 if (pos < 0 || pos > size) {
 printf (" invalid position ! \n");
}    else {
    for(i = pos-1 ; i<size-1; i++) {
     arr[i] = arr [i+1];
     }
     size-- ;
    printf("array after deletion");
    for (i=0; i<size ; i++){
        printf ("%d", arr[i]);
    }
     printf ("\n");
    }
```

17 Deled a value from the last position of an array

```
#include <stdio.h>
    int main (){
int arr[] = {10,20,30,40,50};
  int size = 5;
  int i;
printf ("original array:");
for(i=0; i<size; i++){
  printf ("%d", arr[i]);
}
printf ("\n");
  if (size>0){
    size--;
  }
printf("array after deleting the last element: ");
  for (i=0; i<size; i++){
    printf ("%d", arr[i]);
  }
  printf ("\n");
}
```

18 Delet a value from the array

```
#include <stdio.h>
int deleteElement (int arr[], int size, int value to delet){
    int i, j;
    int found = 0 ;
for (i=0 ; i<size; i++) {
  if (arr[i] == value. to delet) {
      found =1
      break;
    }
    }
if (found) {
  for (i = i ; j<size; j++) {
    arr[i] = arr [j+1] ;
    }
  printf(" element %d deleted successfully \n"
          value to delet ")
  return size-1;
  } else {
printf("elements %d not found in the array
    Value to delete);
  return size;
  }
}
int main() {
```

```
int arr[] = {10,20,30,40,50};
int size = sizeof(arr) / sizeof(arr[0]);
int value_to_delete = 30;
printf("original array: ");
for(int i=0; i<size; i++){
    printf("%d", arr[i]);
}
printf("\n");
size = delet element (arr, size, value to delet);
printf("array of ter deletion: ")
for (int i=0; i<size; i++) {
  printf("%d", arr[i]);
}
printf("\n");
return 0;
}
```

19. Search a value within an array

```
#include <stdio.h>
int main () {
  int arr [] = {10, 20, 30, 40, 50};
  int n = size of (arr) / size of (arr[0]);
  int search value = 30;
  int found = 0;
printf ("Searching for %d in the array..\n",
        search value);
  for (int i=0; i<n; i++) {
    if (arr[i] == search value) {
printf ("value %d found at index %d
        \n", search value, i);
found = 1;
break;
}
}
if (found == 0) {
printf ("value %d not found in the array",
        search value;
}
return 0;
}
```

Section B

1. find out length of a string

```
#include <stdio.h>
#inEIude <string.h>
int main(){
char str[] = "Hello, world!";
int length;
length = strlen(str);
Printf("the length of the string is %d\n",
                                length);
return 0;
}
```

2. Convert a string to lowercase.

```
#include <stdio.h>
#include <ctype.h>
int main (){
char str[] = "Hello WORLD";
Printf("original string: %s\n", str);
for (int i=0; str[i]!='\0'; i++){
str[i] = tolower(str[i]);
}
printf ("lowercase string: %s\n", str);
return 0;
}
```

9 convert a string to uppercase.

```
#include <stdio.h>
#include <ctype.h>
Void to uppercase (char *str) {
    int j=0;
while (str[i] != '\0') {
    str[i] = toupper (str[i]);
        i++;
    }
}
int main () {
    Char my string [100];
printf ("enter a string : ");
    Scanf ("%s", my string);
    to upper case (my string);
Printf (" the uppercase string is : %s \n", mystr
    return 0;
    }
```

4. Convert a String to toggle case

```
#include<stdio.h>
#include<string.h>
#include<ctype.h>
void toggle case (char*str) {
    int i=0;
while (str[i]!='\0') {
if (islower(str[i]) {
    str[i] = toupper (str[i]);
} else if ( is upper (str[i]) {
    str[i]=tolower (str[i]);
    }
    i++;
    }
}
int main () {
char str[100];
ppintf("enter a string :");
gets(str);
toggle case (str);
ppintf(" toggled case string : %.sn", str);
return 0;
}
```

5 Copy one string to another

```
#include <stdio.h>
#include <string.h>
int main () {
char source string [] = " hello, world! " };
char destination. string [20];
strcpy ( destination string , source . string);
ppintf (" source string : %.s \n", source . string );
ppintf (" destination string : %.s \n", destination
                                    string );
};
```

6. Compare two string lexicographically and Printwhich one is greater, smaller, or same.

```
#include <stdio.h>
#include <string.h>
int main () {
  char str1 [100];
  char str 2 [100];
  ppintf ("enter the first string: ");
  scanf ("%.s", str1 );
ppintf ("enter the second string: " );
 scanf ("%.s", str 2 );
int result = strcmp ( str1, str 2 );
if (result > 0 ) {
 ppintf(" the first string is greater than the
                second string \n" );
```

```
} else if (result < 0) {
printf (" the first string is smaller than the second
} else {                                    string \n");
 printf(" the two strings are the same \n");
 }

 return 0;
 }
```

7. Reverse a string

```
#include <stdio.h>
 #include<string.h>
     int main () {
      char str [100];
 char temp;
  int i, j;
 printf ("enter a string : ");
  scanf ("%s", str);
  j = strlen(str)-1;
for(i=0; i<j; i++, j--) {
   temp = str[i];
  str[i] = str[j];
  str[j] = temp;
 }
printf ("Reversed string is : %s\n", str);
  return 0;
}
```

8 Check whether a string is a palindrome

```
#include <stdio.h>
#include <string.h>
int main() {
  char str[100];
  int i, length;
  int is_palindrome = 1;
  printf("enter a string: ");
  scanf("%s", str);
  length = strlen(str);
  for(i=0; i<length/2; i++) {
  if (str[i] != str[length-1-i]) {
      is_palindrome = 0;
      break;
    }
  }
  if (is_palindrome) {
  printf("%s is a palindrome \n", str);
} else {
  printf("%s isnot a palindrome \n", str);
  }
}
```

9 Concatenate one string at the end of another string.

```
#include <stdio.h>
#include <string.h>
int main () {
 Char str1 [100] = "Hello," ;
 Char str2 [ ] = " world!" ;
  strcat (str1, str2);
ppintf ("the concatenated string is : %s \n", str);
 }
```

10. Pprint characters of a string vertically

```
#include <stdio.h>
#include <string.h>
 int main () {
  char str[100];
  pprintf ("enter a string: ");
  Scanf ("%.s", str);
 for (int i=0; str[i] != '\0'; i++) {
  pprintf ("%.c\n", str[i]);
  }
  return 0;
 }
```

11 Print reversed string vertically character by character.

```
#include <stdio.h>
#include <string.h>
int main (){
char str[100];
int length;
printf("enter a string: ");
scanf("%s", str);
length = strlen(str);
for(int i= length-1; i>=0; i--){
printf("%c\n", str[i]);
}
return 0;
}
```

12. Print frequency of each vowel in a given string

```
#include <stdio.h>
#include <string.h>
#include <ctype.h>
int main(){
char str[100];
int vowel count[5]={0};
printf("enter a string: ");
fgets(str, sizeof(str), stdin);
```

```
for (int i=0; str[i] != '\0'; i++){
    char ch = tolower(str[i]);
    switch (ch) {
    case 'a';
    Vowel - Count [0]++;
     break;
    case 'a' :
        Vowel - Count [0]++ ;
          Break;
    Case 'e':
        Vowel Count [1]++ ;
          break;
    Case 'i'
        Vowel - Count [2]++ ;
          break;
    case 'o':
        Vowel - Count [3]++ ;
          break;
    Case 'u'
        Vowel - Count [4]++;
          break;
    }
}
```

```
    printf("Frequency of each vowel : \n");
    printf("a : %d\n", vowel_count[0]);
    printf("e : %d\n", vowel_count[1]);
    printf("i : %d\n", vowel_count[2]);
    printf("o : %d\n", vowel_count[3]);
    printf("u : %d\n", vowel_count[4]);
    return 0;
}
```