PART-C

Write the answer in the space corresponding to the question in the answer sheet. Each sub-question carries +2 marks.

1. Consider the following C-program.

```
#include <stdio.h>

void main ()
{
 static int array[4][4];

  int i, j, m=4, n=4, a;

  /* Manipulate the matrix */
  for (i = 0; i < m; i++)
  {
    a = array[i][i];
    array[i][i] = array[i][m-i-1];
    array[i][m-i-1] = a;
  }

 printf("Elements= %d, %d, %d, %d",
         array[0][0], array[1][0],
         array[2][1], array[3][2]);
}
```

For each of the following initialisations for the matrix `int array[4][4]`, write the output.

(a) `static int array[4][4] =`
`{ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 };`

(b) `static int array[4][4] =`
`{ 1, 5, 9, 10, 2, 6, 8, 6, 7, 3, 1, 2, 8, 4, 5, 4 };`

(c) `static int array[4][4] =`
`{ 5, 6, 4, 3, 1, 8, 9, 10, 4, 5, 4, 6, 3, 2, 1, 1 };`

2. Consider the following C-program.

```
#include <stdio.h>

void func(int array[], int size) {
  int i = 0, j = size-1;

  while (i < j) {
    while (array[i] == 0 && i < j)
      i++;
    while (array[j] == 1 && i < j)
      j--;
    if (i < j) {
      array[i] = 0;
      array[j] = 1;
      i++;
      j--;
    } }
}

int main() {
  int arr[9];
  int array_size = 9, i = 0;

  func(arr, array_size);
  printf("Final array is: ");
  for (i = 0; i < array_size ; i++)
    printf("%d ", arr[i]);
  return 0;
}
```

For each of the following initialisations for the array `int arr[9]`, write the output.

(a) `int arr[9]`
`= {0, 1, 0, 1, 1, 0, 0, 1, 0};`

(b) `int arr[9]`
`= {1, 1, 0, 1, 0, 1, 1, 1, 0};`

(c) `int arr[9]`
`= {1, 0, 1, 0, 1, 0, 1, 0, 1};`