	National University of Computer and Emerging Sciences (Lahore)			
	Course:	**OOP**	**Course code:**	**CS217**
	Section:	**BSCS-2B**	**Semester:**	**Spring 2024**
	Duration:	**20 minutes**	**TotalMarks:**	**10**
	Date:		**ID:**	**B**
	Name:		**Roll no:**	

Question 1: Give the output of the following code snippet. Identify syntax or logical errors, if present in the code, and suggest the required correction.

```
void Print(int * arr, int size)
{
   cout<<"array is:";
   for(int i = 0; i < size; i++)
   { cout << arr[i] << ","; }
   cout << endl;
}
int main()
{
   int* arr = new int[6];
   arr[2] = 4; arr[0] = 2;
   arr[3] = 8; arr[1] = 1;
   arr[4] = 5; arr[5] = 3;
   Print(arr,6);
   for (int i = 0; i < 3; i++)
   {
      int temp = *(arr + 2 * i);
      *(arr + 2 * i) = *(arr + 2 * i + 1);
      *(arr + 2 * i + 1) = temp;
   }
   Print(arr,6);
}
```

Output:

```
array is: 2,1,4,8,5,3,
array is: 1,2,8,4,3,5,
```

Error (if any):

Array is not deallocated (memory leakage).
Delete[] arr;

Question 2: Write C++ functions *"initialize()"* and *"updateArray()"* . initialize() accepts an int pointer and size and allocates an array of size dynamically. updateArray() accepts two arguments; an int pointer to an array and the size of array. The function should then reverse the order of elements in the given array.

NOTE: you are not allowed to use indexing with subscript operator "[ ]" for updateArray()

```
void initialize(int *& ptr, int size)
{
  ptr = new int[size];

  // Check if memory allocation was successful
  if (arr_ptr == nullptr)
  {
     cout << "Memory allocation failed!" << endl;
     return;
  }
}
```

```
void updateArray(int arr[], int size)
{
   // Swap elements until they meet in the middle.
   for (int i = 0; i < size / 2; i++)
   {
      int temp = arr_ptr[i];
      arr_ptr[i] = arr_ptr[size - i - 1];
      arr_ptr[size - i - 1] = temp;
   }
}
```