Question 1 (Passing parameters):

```cpp
#include <iostream>

void swap(int* a, int b, int c) {
  int temp = a[0];
  a[0] = a[1];
  a[1] = temp;
  temp = b;
  b = c;
  c = temp;
}

using namespace std;
int main() {
  int arr[] = {1,2,3,4,5,6,7,8,9,10};
  int b = 4, c = 5, d = 6;
  swap(arr, b, c);
  arr[++d] = d++;
  for (int i = 0; i < 10; i++)
  cout << arr[i] << " ";
  cout << endl;
  cout << b << " " << c << " " << d;
}
```

Please draw the memory content of the array `arr` and variables `b,c,d` after running the code.

Solutions Answers:

Variable	Hex value
`arr[0..9]`	$2, 1, 3, 4, 5, 6, 7, 7, 9, 10$
b	4
c	5
c	8

`swap()` can exchange two values `arr[0]` and `arr[1]`, because it receives the pointer to the whole array. (Therefore, `arr[0]=2`, `arr[1]=1`.) On the other hand, very similar code in `swap()` fails to swap variables `b` and `c`, since they are passed by value.

The line `arr[++d] = d++;` starts evaluating the expression from the right-hand side (`dd++` makes $d = 7$). Then it performs the pre-increment on d (it becomes $d = 8$) and assigns $arr[7]$ to be 7. it just assigns incremented value d (it is 7) to the element `arr[6]`, which is already equal to 7.