Lab 3 – Arrays – Suggested Answers

Q1: Suggested Answer

```
int findAr1D(int size, int ar[], int target)
{
  int j;
  for (j = 0; j < size; j++)
    if (ar[j] == target)
      return j;
  return -1;
}
/* another version */
/*
int findAr1D(int size, int ar[], int target)
{
  int j;
  for (j = 0; j < size; j++)
    if (*(ar+j) == target)
      return j;
  return -1;
}
*/
```

Q2: Suggested Answer

```
void swap2Rows(int ar[][SIZE], int r1, int r2)
/* swaps row ar[r1] with row ar[r2] */
{
  int temp;
  int n;

  for (n = 0; n < SIZE; n++) {
    temp = ar[r1][n] ;
    ar[r1][n] = ar[r2][n];
    ar[r2][n] = temp;
  }
}
void swap2Cols(int ar[][SIZE], int c1, int c2)
/* swaps column ar[][c1] with column ar[][c2] */
{
  int temp;
  int n;

  for (n = 0; n < SIZE; n++) {
    temp = ar[n][c1] ;
    ar[n][c1] = ar[n][c2];
    ar[n][c2] = temp;
  }
```

```
}
```

Q3: Suggested Answer

```
void printReverse1(int ar[], int size)
{
  int i;
  printf("printReverse1(): ");
  if (size > 0) {
    for (i=size-1; i>=0; i--)
      printf("%d ", ar[i]);
  }
  printf("\n");
}
void printReverse2(int ar[], int size)
{
  int i;
  printf("printReverse2(): ");
  if (size > 0) {
    for (i=size-1; i>=0; i--)
      printf("%d ", *(ar+i));
  }
  printf("\n");
}
/* reverseAr reverses the array contents and passes that back to the calling function */
void reverseAr1D(int ar[], int size)
{
  int i, temp;
  if (size > 0) {
    for (i=0; i<size/2; i++){
      temp = ar[i];
      ar[i] = ar[size-i-1];
      ar[size-i-1] = temp;
    }
  }
}
```