Part A:

1. Give two separate statements that assign the starting address of array “numbers” to the pointer variable “nPtr”.
   1. Int \*nPtr = &numbers[0];
   2. nPtr = numbers;
2. Print the elements of array numbers using pointer/offset notation with pointer nPtr.
   1. cout << \*(nPtr + i) << endl;
3. Print the elements of array “numbers” by subscripting point “nPtr”.
   1. cout << \*(nPtr + i) << endl; **//inside a for loop**
4. Assume that the variable “ptr” is of type **char** \* and arrays s1[100] and s2[100] are of type **char.**
   1. s2 = s1;
   2. ptr = &s2[0];
5. Assume address c1 is 100 and address c2 is 2000, what is the output printed on the sceen?
   1. “\*p2 = a”
   2. “\*p2 = A, p2 = A”
   3. “\*p3 = A, p3 = A”
6. Assume that unsigned integers are stored in 2 bytes and starting address of array values is at location 2500. Answer the following questions related to array values.
   1. unsigned int values[SIZE]= {2, 3, 4, 5, 6};
   2. unsigned int \*ptr;
   3. cout << values[i] << endl; **//inside of for loop**
   4. ptr = values;
   5. cout << \*(ptr + i) << endl; **//inside of for loop**
   6. 2506 **//since its two bytes**
   7. 2500