Q1:

*** A = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10}**

*** B = { 10, 9, 8, 7, 6, 5, 4, 3, 2, 1 }**

*** C = {5, 2, 13, 9, 1, 7, 6, 8, 1, 15, 4, 11}**

*** D = {'S', 'B', 'I', 'M', 'H', 'Q', 'C', 'L', 'R', 'E', 'P', 'K'}**

A) = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10}

Shell Sort

```
gap: 5 , try to 5 sorted array
no swap

gap: 2 , try to 2 sorted array
no swap

gap: 1 , try to 1 sorted array
no swap

it is already sorted

After sorted array:
1 2 3 4 5 6 7 8 9 10
```

B) = { 10, 9, 8, 7, 6, 5, 4, 3, 2, 1 }

Shell Sort

```
gap: 5 , try to 5 sorted array

 Before compare and swap:
 10 9 8 7 6 5 4 3 2 1
 compare and swap: index: 5 value: 5 with index: 0 value: 10
 After compare and swap:
 5 9 8 7 6 10 4 3 2 1

 Before compare and swap:
 5 9 8 7 6 10 4 3 2 1
 compare and swap: index: 6 value: 4 with index: 1 value: 9
 After compare and swap:
 5 4 8 7 6 10 9 3 2 1

 Before compare and swap:
 5 4 8 7 6 10 9 3 2 1
 compare and swap: index: 7 value: 3 with index: 2 value: 8
 After compare and swap:
 5 4 3 7 6 10 9 8 2 1

 Before compare and swap:
 5 4 3 7 6 10 9 8 2 1
 compare and swap: index: 8 value: 2 with index: 3 value: 7
 After compare and swap:
 5 4 3 2 6 10 9 8 7 1
```

```
Before compare and swap:
5 4 3 2 6 10 9 8 7 1
compare and swap: index: 9 value: 1 with index: 4 value: 6
After compare and swap:
5 4 3 2 1 10 9 8 7 6

gap: 2 , try to 2 sorted array

Before compare and swap:
5 4 3 2 1 10 9 8 7 6
compare and swap: index: 2 value: 3 with index: 0 value: 5
After compare and swap:
3 4 5 2 1 10 9 8 7 6

Before compare and swap:
3 4 5 2 1 10 9 8 7 6
compare and swap: index: 3 value: 2 with index: 1 value: 4
After compare and swap:
3 2 5 4 1 10 9 8 7 6

Before compare and swap:
3 2 5 4 1 10 9 8 7 6
compare and swap: index: 4 value: 1 with index: 2 value: 5
compare and swap: index: 2 value: 1 with index: 0 value: 3
After compare and swap:
1 2 3 4 5 10 9 8 7 6

Before compare and swap:
1 2 3 4 5 10 9 8 7 6
compare and swap: index: 7 value: 8 with index: 5 value: 10
After compare and swap:
1 2 3 4 5 8 9 10 7 6

Before compare and swap:
1 2 3 4 5 8 9 10 7 6
compare and swap: index: 8 value: 7 with index: 6 value: 9
After compare and swap:
1 2 3 4 5 8 7 10 9 6

Before compare and swap:
1 2 3 4 5 8 7 10 9 6
compare and swap: index: 9 value: 6 with index: 7 value: 10
compare and swap: index: 7 value: 6 with index: 5 value: 8
After compare and swap:
1 2 3 4 5 6 7 8 9 10

gap: 1 , try to 1 sorted array
no swap
it is already sorted

After sorted array:
1 2 3 4 5 6 7 8 9 10
```

C) = {5, 2, 13, 9, 1, 7, 6, 8, 1, 15, 4, 11}

Shell Sort

```
gap: 6 , try to 6 sorted array

Before compare and swap:
5 2 13 9 1 7 6 8 1 15 4 11
compare and swap: index: 8 value: 1 with index: 2 value: 13
After compare and swap:
5 2 1 9 1 7 6 8 13 15 4 11
```

```
gap: 2 , try to 2 sorted array

Before compare and swap:
5 2 1 9 1 7 6 8 13 15 4 11
compare and swap: index: 2 value: 1 with index: 0 value: 5
After compare and swap:
1 2 5 9 1 7 6 8 13 15 4 11

Before compare and swap:
1 2 5 9 1 7 6 8 13 15 4 11
compare and swap: index: 4 value: 1 with index: 2 value: 5
After compare and swap:
1 2 1 9 5 7 6 8 13 15 4 11

Before compare and swap:
1 2 1 9 5 7 6 8 13 15 4 11
compare and swap: index: 5 value: 7 with index: 3 value: 9
After compare and swap:
1 2 1 7 5 9 6 8 13 15 4 11

Before compare and swap:
1 2 1 7 5 9 6 8 13 15 4 11
compare and swap: index: 7 value: 8 with index: 5 value: 9
After compare and swap:
1 2 1 7 5 8 6 9 13 15 4 11

Before compare and swap:
1 2 1 7 5 8 6 9 13 15 4 11
compare and swap: index: 10 value: 4 with index: 8 value: 13
compare and swap: index: 8 value: 4 with index: 6 value: 6
compare and swap: index: 6 value: 4 with index: 4 value: 5
After compare and swap:
1 2 1 7 4 8 5 9 6 15 13 11

Before compare and swap:
1 2 1 7 4 8 5 9 6 15 13 11
compare and swap: index: 11 value: 11 with index: 9 value: 15
After compare and swap:
1 2 1 7 4 8 5 9 6 11 13 15

gap: 1 , try to 1 sorted array

Before compare and swap:
1 2 1 7 4 8 5 9 6 11 13 15
compare and swap: index: 2 value: 1 with index: 1 value: 2
After compare and swap:
1 1 2 7 4 8 5 9 6 11 13 15

Before compare and swap:
1 1 2 7 4 8 5 9 6 11 13 15
compare and swap: index: 4 value: 4 with index: 3 value: 7
After compare and swap:
1 1 2 4 7 8 5 9 6 11 13 15

Before compare and swap:
1 1 2 4 7 8 5 9 6 11 13 15
compare and swap: index: 6 value: 5 with index: 5 value: 8
compare and swap: index: 5 value: 5 with index: 4 value: 7
After compare and swap:
1 1 2 4 5 7 8 9 6 11 13 15

Before compare and swap:
1 1 2 4 5 7 8 9 6 11 13 15
compare and swap: index: 8 value: 6 with index: 7 value: 9
compare and swap: index: 7 value: 6 with index: 6 value: 8
compare and swap: index: 6 value: 6 with index: 5 value: 7
After compare and swap:
1 1 2 4 5 6 7 8 9 11 13 15

After sorted array:
1 1 2 4 5 6 7 8 9 11 13 15
```

D) = {‘S’, ‘B’, ‘I’, ‘M’, ‘H’, ‘Q’, ‘C’, ‘L’, ‘R’, ‘E’, ‘P’, ‘K’}

Shell Sort

```
gap: 6 , try to 6 sorted array

 Before compare and swap:
 S B I M H Q C L R E P K
 compare and swap: index: 6 value: C with index: 0 value: S
 After compare and swap:
 C B I M H Q S L R E P K

 Before compare and swap:
 C B I M H Q S L R E P K
 compare and swap: index: 9 value: E with index: 3 value: M
 After compare and swap:
 C B I E H Q S L R M P K

 Before compare and swap:
 C B I E H Q S L R M P K
 compare and swap: index: 11 value: K with index: 5 value: Q
 After compare and swap:
 C B I E H K S L R M P Q

 gap: 2 , try to 2 sorted array

 Before compare and swap:
 C B I E H K S L R M P Q
 compare and swap: index: 4 value: H with index: 2 value: I
 After compare and swap:
 C B H E I K S L R M P Q

 Before compare and swap:
 C B H E I K S L R M P Q
 compare and swap: index: 8 value: R with index: 6 value: S
 After compare and swap:
 C B H E I K R L S M P Q

 Before compare and swap:
 C B H E I K R L S M P Q
 compare and swap: index: 10 value: P with index: 8 value: S
 compare and swap: index: 8 value: P with index: 6 value: R
 After compare and swap:
 C B H E I K P L R M S Q

 gap: 1 , try to 1 sorted array

 Before compare and swap:
 C B H E I K P L R M S Q
 compare and swap: index: 1 value: B with index: 0 value: C
 After compare and swap:
 B C H E I K P L R M S Q

 Before compare and swap:
 B C H E I K P L R M S Q
 compare and swap: index: 3 value: E with index: 2 value: H
 After compare and swap:
 B C E H I K P L R M S Q

 Before compare and swap:
 B C E H I K P L R M S Q
 compare and swap: index: 7 value: L with index: 6 value: P
 After compare and swap:
 B C E H I K L P R M S Q

 Before compare and swap:
 B C E H I K L P R M S Q
 compare and swap: index: 9 value: M with index: 8 value: R
 compare and swap: index: 8 value: M with index: 7 value: P
 After compare and swap:
 B C E H I K L M P R S Q
```

```
Before compare and swap:
B C E H I K L M P R S Q
compare and swap: index: 11 value: Q with index: 10 value: S
compare and swap: index: 10 value: Q with index: 9 value: R
After compare and swap:
B C E H I K L M P Q R S

After sorted array:
B C E H I K L M P Q R S
```