1. Implement union. Take inputs from the user and make a union array of int/char.

Solution:

```cpp
#include<iostream>
using namespace std;

union Uni{
    int i;
    char c;
};

int main() {
    union Uni arr[100];
    int tags[100];
    int n;
    cout<<"Enter number of elements: ";
    cin>>n;
    for (int i=0; i<n; i++){
        cout<<"Press 1 for character and 0 for number: ";
        cin>>tags[i];
        cout<<"Enter Number or Character: ";
        if (tags[i] == 0){
            cin>>arr[i].i;
        } else {
            cin>>arr[i].c;
        }
    }
    for (int i=0; i<n; i++) {
        if (tags[i] == 0){
            cout<<arr[i].i<<" ";
        }else {
            cout<<arr[i].c<<" ";
        }
    }
    return 0;
}
```

```
Enter number of elements: 7
Press 1 for character and 0 for number: 0
Enter Number or Character: 46
Press 1 for character and 0 for number: 0
Enter Number or Character: 65
Press 1 for character and 0 for number: 1
Enter Number or Character: A
Press 1 for character and 0 for number: 1
Enter Number or Character: H
Press 1 for character and 0 for number: 1
Enter Number or Character: U
Press 1 for character and 0 for number: 0
Enter Number or Character: 34
Press 1 for character and 0 for number: 0
Enter Number or Character: 87
46 65 A H U 34 87
Process returned 0 (0x0)    execution time : 28.041 s
Press any key to continue.
```

2. Ask user whether the number should be in beginning or end and take the number. Then arrange it in an array.

Solution:

```cpp
#include<iostream>
using namespace std;


int main() {
    int arr[100];
    int n, choice;
    cout<<"Enter number of elements: ";
    cin>>n;
    int l=0;
    int h=n-1;
    for (int i=0; i<n; i++){
        cout<<"Press 1 for end and 0 for beginning: ";
        cin>>choice;
```

```cpp
        cout<<"Enter Number: ";
        if (choice == 0){
            cin>>arr[l];
            l++;
        } else {
            cin>>arr[h];
            h--;
        }
    }
    for (int i=0; i<n; i++) {
        cout<<arr[i]<<" ";
    }
    return 0;
}
```

```
Enter number of elements: 7
Press 1 for end and 0 for beginning: 0
Enter Number: 12
Press 1 for end and 0 for beginning: 0
Enter Number: 34
Press 1 for end and 0 for beginning: 1
Enter Number: 56
Press 1 for end and 0 for beginning: 1
Enter Number: 63
Press 1 for end and 0 for beginning: 1
Enter Number: 37
Press 1 for end and 0 for beginning: 0
Enter Number: 55
Press 1 for end and 0 for beginning: 0
Enter Number: 89
12 34 55 89 37 63 56
Process returned 0 (0x0)    execution time : 31.265 s
Press any key to continue.
```

3. Implement Union ask user the input and its type. Then if it's an integer place it from beginning and if it's a character place it from end.

Solution:

```cpp
#include<iostream>
using namespace std;

union Uni{
    int i;
    char c;
};

int main() {
    union Uni arr[100];
    int tags[100];
    int n, choice;
    cout<<"Enter number of elements: ";
    cin>>n;
    int l=0;
    int h = n-1;
    for (int i=0; i<n; i++){
        cout<<"Press 1 for character and 0 for number: ";
        cin>>choice;
        cout<<"Enter Number or Character: ";
        if (choice == 0){
            cin>>arr[l].i;
            tags[l] = 0;
            l++;
        } else {
            cin>>arr[h].c;
            tags[h] = 1;
            h--;
        }
    }
    for (int i=0; i<n; i++) {
        if (tags[i] == 0){
            cout<<arr[i].i<<" ";
        }else {
            cout<<arr[i].c<<" ";
        }
    }
```

```
    }
    return 0;
}
```

```
Enter number of elements: 7
Press 1 for character and 0 for number: 0
Enter Number or Character: 34
Press 1 for character and 0 for number: 0
Enter Number or Character: 58
Press 1 for character and 0 for number: 1
Enter Number or Character: g
Press 1 for character and 0 for number: 1
Enter Number or Character: A
Press 1 for character and 0 for number: 0
Enter Number or Character: 70
Press 1 for character and 0 for number: 1
Enter Number or Character: H
Press 1 for character and 0 for number: 1
Enter Number or Character: I
34 58 70 I H A g
Process returned 0 (0x0)    execution time : 30.037 s
Press any key to continue.
```