



# **CS10003:** **Programming & Data Structures**

**Dept. of Computer Science & Engineering**  
**Indian Institute of Technology Kharagpur**

*Autumn 2020*

## Alternate Version 3

**Define an array  
of  
large size and  
use  
only the  
required  
number of  
elements**

```
int main()
{
    int a[100], i, min, n;

    scanf ("%d", &n); /* Number of elements */
    for (i=0; i<n; i++)
        scanf ("%d", &a[i]);

    min = a[0];
    for (i=1; i<n; i++)
    {
        if (a[i] < min)
            min = a[i];
    }
    printf ("\n Minimum is %d", min);
    return 0;
}
```

## Example 2: Computing cgpa

**Handling two  
arrays  
at the same  
time**

```
const int nsub = 6;

int main()
{
    int grade_pt[nsub], cred[nsub], i,
        gp_sum=0, cred_sum=0;
    double gpa;

    for (i=0; i<nsub; i++)
        scanf ("%d %d", &grade_pt[i], &cred[i]);

    for (i=0; i<nsub; i++)
    {
        gp_sum += grade_pt[i] * cred[i];
        cred_sum += cred[i];
    }
    gpa = ((double) gp_sum) / cred_sum;
    printf ("\n Grade point average: is %.2lf", gpa);
    return 0;
}
```

# Example: Selection Sort

- Sort the elements of an array  $A$  with  $n$  elements in ascending order
- Basic Idea:
  - Find the min of the  $n$  elements, swap it with  $A[0]$  (so min is at  $A[0]$  now)
  - Now find the min of the remaining  $n-1$  elements, swap it with  $A[1]$  (so 2<sup>nd</sup> min is at  $A[1]$  now)
  - Continue until no more elements left

```

int main() {
    int A[100], n, i, j, k, min, pos, temp;
    scanf("%d", &n);
    for (i=0; i<n; ++i) scanf("%d", &A[i]);
    for (i = 0; i < n - 1; ++i) {
        min = A[i]; pos = i;
        for (j = i + 1; j < n; ++j) {
            if (A[j] < min) {
                min = A[j];
                pos = j;
            }
        }
        temp = A[i];
        A[i] = A[pos];
        A[pos] = temp;
        for (k=0; k<n; ++k) printf("%d ", A[k]);
        printf("\n");
    }
    return 0;
}

```

# Output

```
6
7 12 5 15 17 9
5 12 7 15 17 9
5 7 12 15 17 9
5 7 9 15 17 12
5 7 9 12 17 15
5 7 9 12 15 17
```

```
8
9 8 7 6 5 4 3 2
2 8 7 6 5 4 3 9
2 3 7 6 5 4 8 9
2 3 4 6 5 7 8 9
2 3 4 5 6 7 8 9
2 3 4 5 6 7 8 9
2 3 4 5 6 7 8 9
2 3 4 5 6 7 8 9
```

# Things you cannot do

- You cannot
  - use = to assign one array variable to another  
a = b; /\* a and b are arrays \*/
  - use == to directly compare array variables  
if (a == b) .....