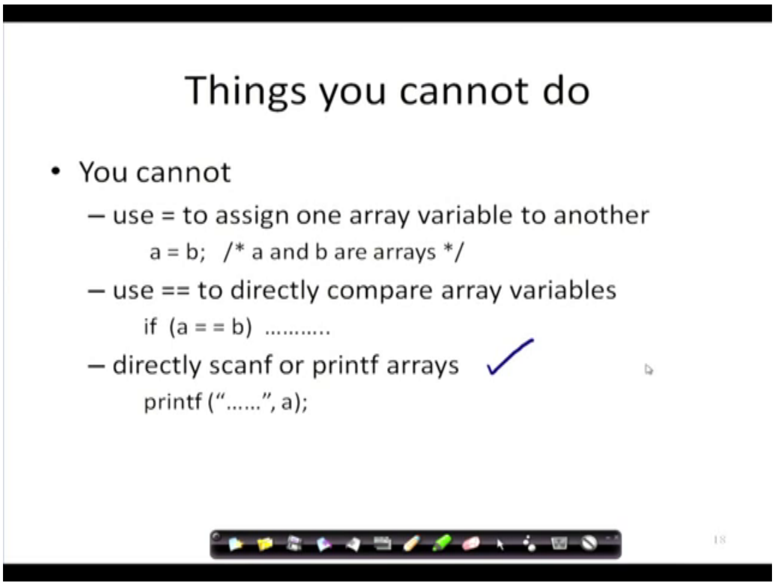
Problem Solving through Programming in C

Prof. Anupam Basu
Department of Computer Science & Engineering
Indian Institute of Technology, Kharagpur

Lecture – 30
Array Problem

We were discussing about arrays and we have seen how we can read an array, we have seen how we can print an array, we have also seen how we can access the different elements of an array using for loops and while loops.

(Refer Slide Time: 00:44)

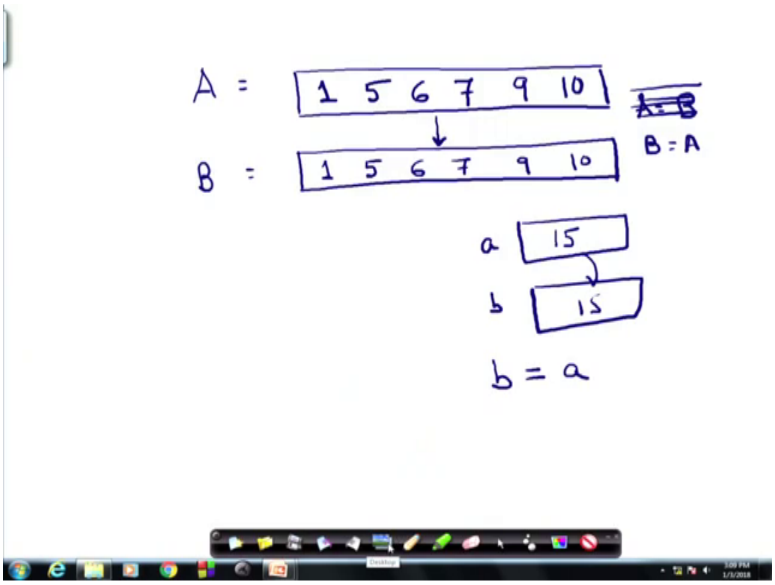


Things you cannot do

- 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) ……….
  - directly scanf or printf arrays ✓
    printf (“……”, a);

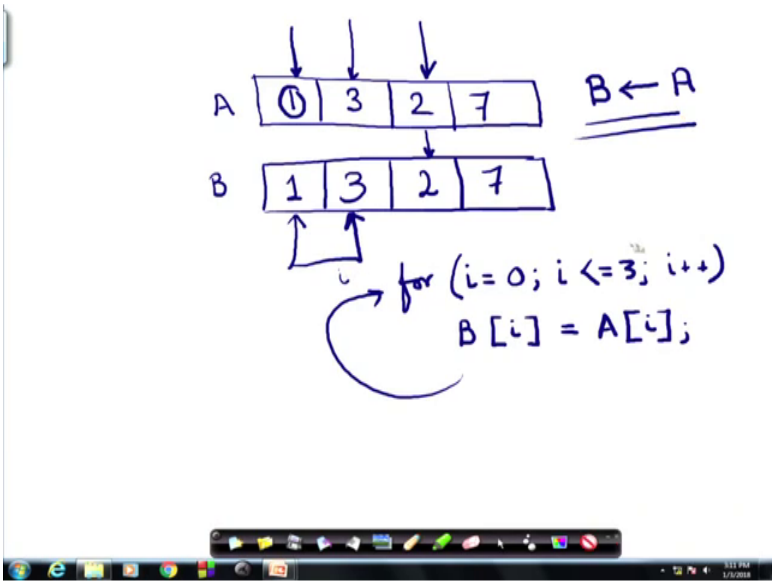
Now, couple of things that we cannot do using an array are as you can see here we cannot use. These equalities, this assignment for assigning one array variable to another, for example, if I have, suppose I have one array A suppose I have one array A which is this array having some elements suppose it is an integer array and the elements here are 1 5 6 7 9 10.

(Refer Slide Time: 01:05)



Now, and there is another array B like this, I want that this data be transferred to 1 to this array B as 1 5 6 7 9 10. Now, that I cannot do using A assigned B, I cannot do this or B assigned A, I am sorry this is wrong using B assigned a i cannot do that. However, it is true that A is also a variable of type array and B is also a variable of type array. But for other variable types for example, if A let us say A is a integer variable and B is another integer variable and a was 15 and then I could have assigned B assigned A. So, B will become 15, but this sort of straightforward assignment cannot be done in the case of an array. What we have to do in order to, this such in order to do such assignments in an array is that we will have to do it component by component element by element.

(Refer Slide Time: 03:07)

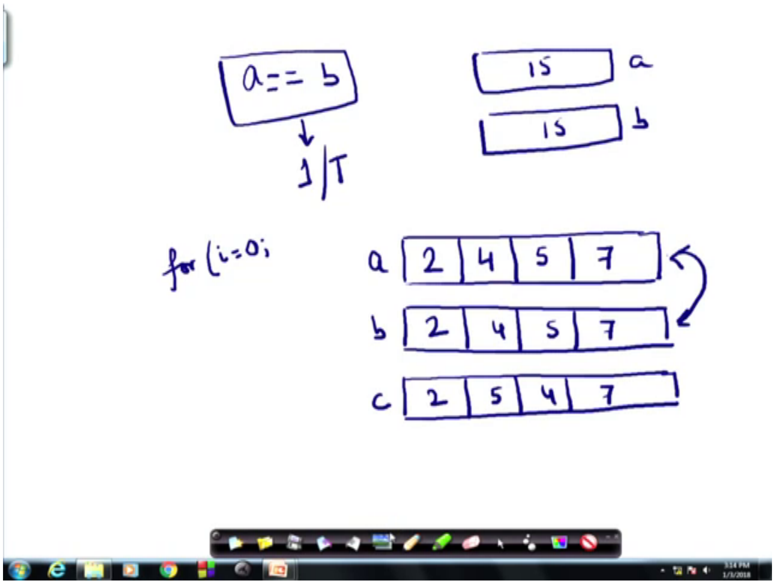


For example, I can do this let me again take another array A. Now, this time I take an array A to be a little smaller of say 4 elements 1 3 2 7 and there is another element B and I actually intend my intention is that the data in A goes to the array B. Now, as I said for an array I have to do it element by element. So, suppose this there are 4 elements in the array and I write a for loop for i equals 0, i less than equal to 3, i plus plus. B i is assigned A i what will happen through this? A i, i is 0. So, A i; that means, 0 this element will be taken and that will be copied to the 0th element of B here, both of them are i, so that 0. So, this will become 1.

Next i is incremented, so i becomes 1. So, this element is being pointed at and I transfer this element to the corresponding element this will also be incremented i, this is also i, so i is 1. So, this will be copied here similarly for this element next in this loop I will be incremented and this will be loaded here. Similarly, for the last element it will go on. Therefore, ultimately I will get the same thing 1 3 2 7 as was my intention, but I cannot do it directly I have to do it in the form of a loop, element by element I have to transfer them all right. So, we cannot use this assignment directly. What else am I prohibited to do. So, A and B are two arrays I cannot do this.

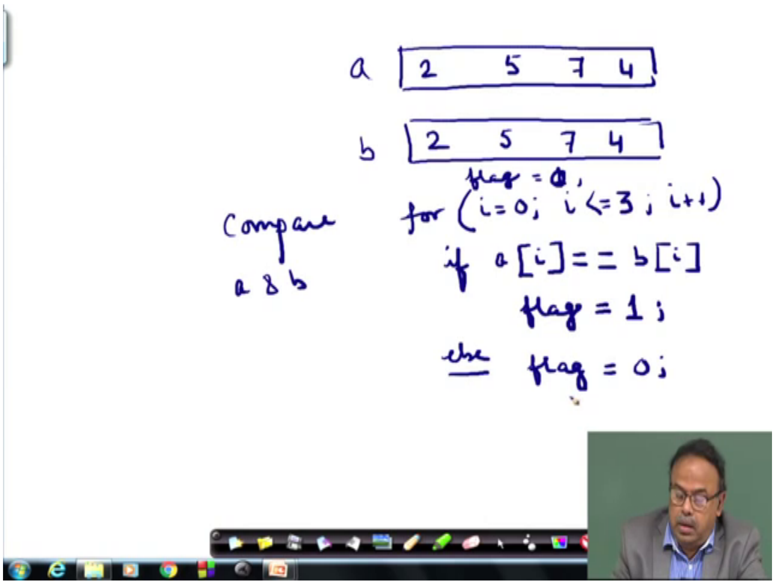
Also I cannot compare two arrays using this sort of equality checker. Earlier we have seen that I can check whether two elements are equal logical operation I could do it using normal a equals b, I am sorry a is b. If a was a variable 15 and b was another variable 15 in that case a equality checker b will lead to a value 1 or 2. But if a is an array and b is another array I cannot do it directly in that way.

(Refer Slide Time: 06:12)



There also I can write a piece of program in which I can check them element by element. So, please note suppose I have got an array a which is 2 4 5 7. Another array b is 2 4 5 7 then these two arrays are equal. However, if I have an array with the same elements, but in a different arrangement say for example, 2 5 4 7 these two l this a and c are not the same therefore, how can I compare how can I compare these two arrays how can I compare them. In order to compare these two arrays a and b what I need to do is I have to again write a small loop like for let me do it clearly.

(Refer Slide Time: 08:19)



a [2 5 7 4]

b [2 5 7 4]

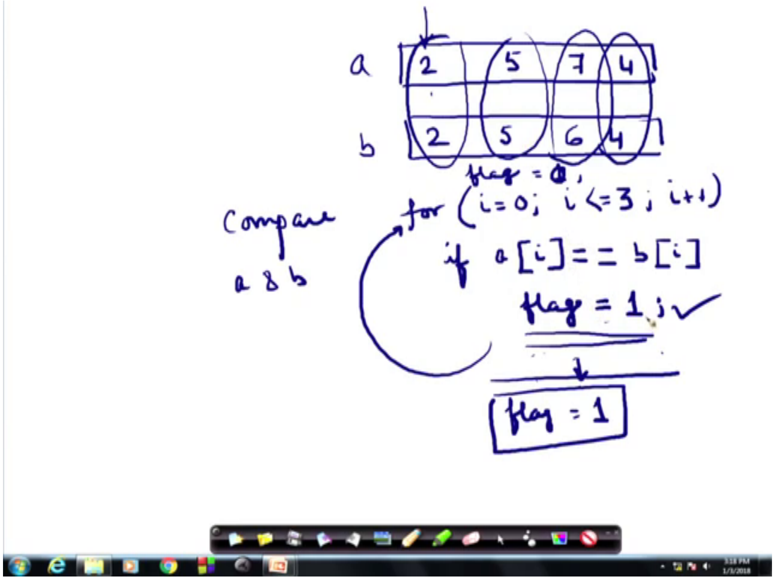
Compare a & b

```
flag = 0;
for (i=0; i<=3; i++)
    if a[i]==b[i]
        flag = 1;
    else flag = 0;
```

Say I have got an array 2 5 7 4 another array 2 5 7 4, a and b are the equal I can compare them I want to compare them compare a and b. Now, what can I do? I can do it in this way for i 0, i less than equal to 3 because there are 4 elements in the array, i plus plus. Suppose I put a flag here a variable flag equals 1, suppose I keep a flag variable I have initialize it to 1 all right. Now, for i equal to 1 to 0 or let me make it simpler I keep it 0 initially this is 0.

So, now I do if a i is equal to b i flag is assigned 1 and flag is assigned 1, else flag assigned 0 and I come out of this. So, what I am trying to do is here let me let me the avoid this part for the time being let us see we will develop it gradually. What will happen here if this is done let us see, for if I do this piece if I run this piece of code then what will happen. For i equal to 0; that means, I will be comparing with this.

(Refer Slide Time: 10:47)



```
flag = 0;
for (i=0; i<=3; i++)
    if a[i] == b[i]
        flag = 1;
```

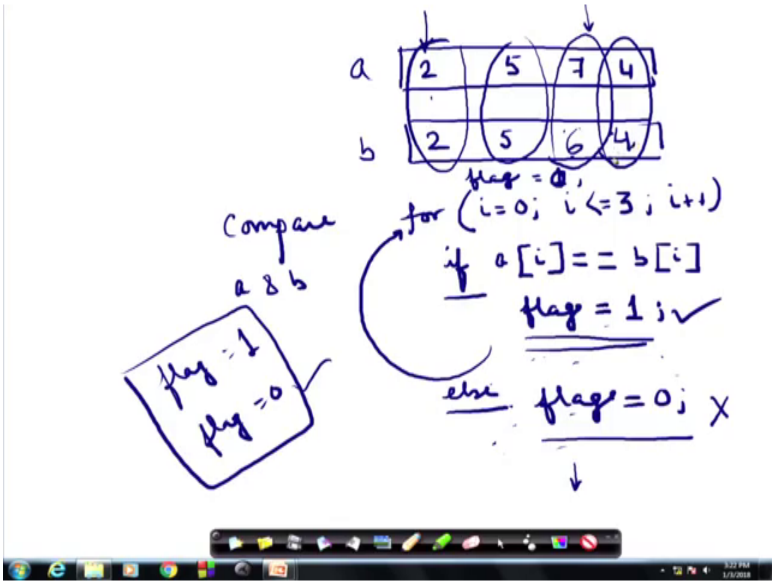
And if a i is equal to a i; that means, I am comparing these two if these two are equal then flag is 1; that means, till. Now, it is one, but this will go on here, so here 1. Now, 5 and 5 is 1 fine, 7 and 7 is compared again it is written 1, 4 and 4 again it is written 1. So, it has been done and ultimately come out with the value of flag ultimately I come out when I come out of this the value of flag is 1 and so I can say if flag is one then they are equal. But there is some problem with this program what is that? First of all every time I was assigning flag to be 1, that is one problem.

Now, suppose another problem is suppose this value is 6 then what will happen let us trace this once again 2 and 2 will match. So, the value of flag will be 1 fine. 5 and 5 will be matched, again I am not bothering about how many times unnecessarily I am writing into flag that apart flag will be 1. Now, 7 and 6 they are not equal to 1. So, I will come out of this if statement, but will go again in the loop. So, what should I do here? If these are this is not equal, not equal then I will have to set flag to 0 if there is a mismatch. So, if I do it like this then what will happen? It is this part, if a i is equal to b i then flag will be one and else; that means, with this else flag will be 0.

So, ultimately now, suppose, but here again there is a problem what will happen. So, 7 and 6 are not matching. So, this is not true. So, flag will be 0 and then I will again go into the loop because the loop is loop has not yet ended and here I will find that these two elements are equal 4 and 4. So, it will come at this point and flag will be 1 and then I come out of the loop I will come out with flag equal to 1, but these two arrays are not equal because I can see that the elements are different.

So, this problem this program will not solve my purpose. I leave it to you for a while to see how to think how you can solve this problem you need to apply your mind a little bit to write this piece of program which will simply check whether two arrays are equal or not if they are equal. That means, if the elements are the same at every position corresponding position the elements are same in that case it will come out with a flag value is 1, otherwise it will come out of the flag value of 0, that is the task. I hope you understand where the problem is in this problem, in this program.

(Refer Slide Time: 14:31)

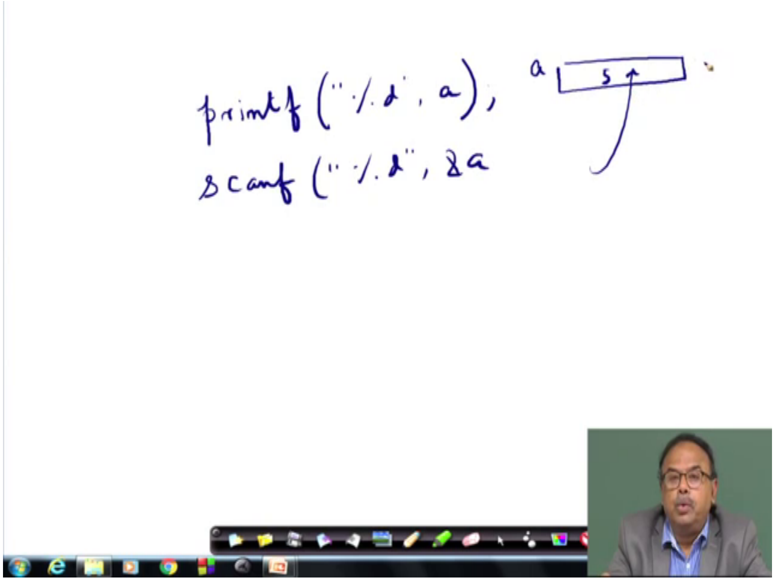


```
flag = 0;
for (i=0; i<=3; i++)
    if a[i] == b[i]
        flag = 1;
    else flag = 0;
```

You see here had the mismatch been at the last point last moment, at the last element 7 6 then it would have worked fine. So, that is why at the beginning we said this is not just C programming, it is learning to write programs using logic. Suppose this. this was 7 and this was 3 then the errors are not equal, array are not equal, here flag would be 1, here flag would be 1, here flag would be 1, here flag would be 0 and I had come out and my answer would be correct by chance. The answer is correct by chance because as we have seen that here if I have an error if I have a mismatch here and suppose here the things are all right then I would once gate the flag is 0, but next time that flag will be made to 1 and ultimately when I come out I will have the flag value to be 1. So, the correct answer has eluded me. So, you just think of how I can modify this program such that I get the correct result as I intended.

So, we cannot use these directly, directly it compare the array variables I cannot compare. What else? So, if a assigned b this sort of thing I cannot do. Also the other thing that I cannot do is scanf and printf statements I cannot read the array in one shot that is not allowed; scanf and printf I could normally, normally what could I do.

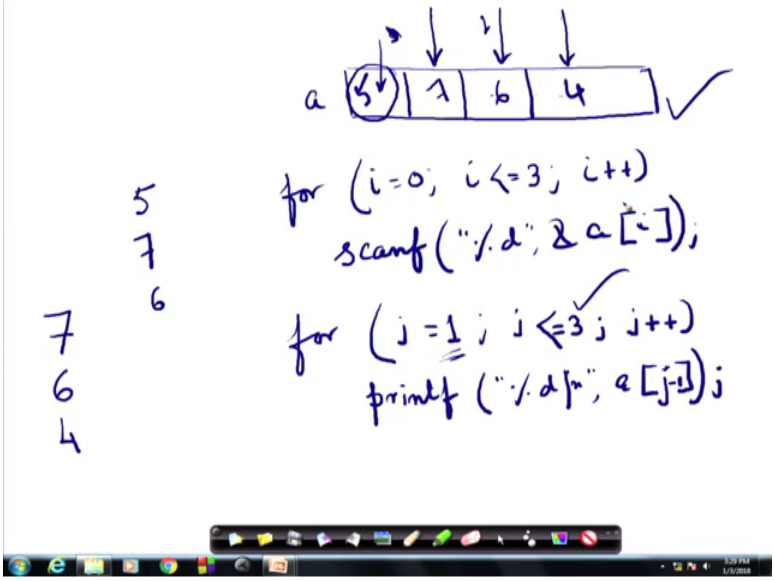
(Refer Slide Time: 17:13)



```
printf ("%d", a);
scanf ("%d", &a
```

Normally if I have a variable a sum variable value 5 I could have simply done printf percentage d a. So, this could be printed. Similarly I could have done scanf, percentage d and they I could have read a value here. But if it is an array I cannot do it in this way we already know how we can do that. We know that in order to read an array what we need to do is read it again in a loop in a for loop element by element so, it can be for i assign 0, i less than equal to whatever 3, i plus plus, scanf percentage d and a i.

(Refer Slide Time: 17:51)



```
for (i=0; i<=3; i++)
    scanf("%d", &a[i]);
for (j=1; j<=3; j++)
    printf("%d\n", a[j-1]);
```

So, this array a will be red element by element I equal to 0. So, first this element will be red same might be 5, whatever element is 5 there might be 7 etcetera in that way I can read them element by element. Similarly for printing an array 6 4, for printing an array I can simply again do that for say j assign 0, j let me change a little bit, is any trouble here j less than 3, j plus plus, printf percentage d a i. Here of course, I do not give the scanf again.

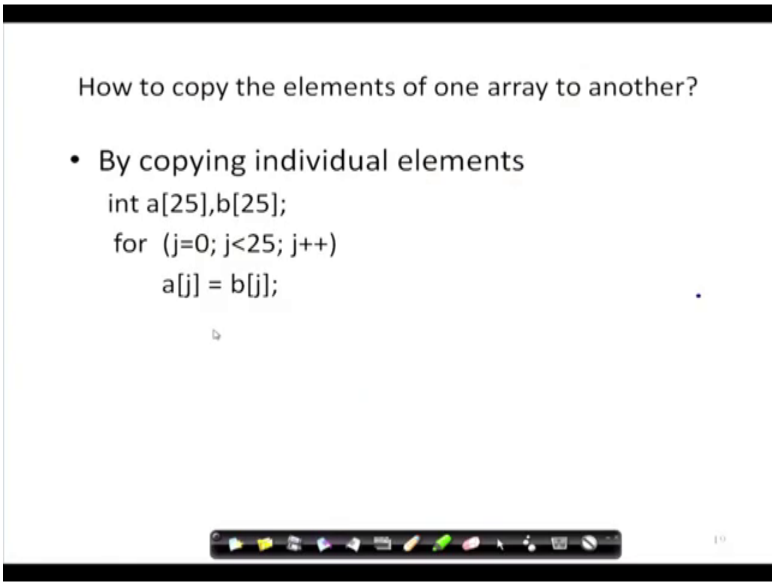
Now, is any problem with this? Here I made i less than equal to 3, here I am making j less than 3 for the same size of array you can think of, you can look at it this is also correct because here I am starting with 1. So, 1 I am looking at. Now, there is a mistake here, I have done a couple of mistakes. What will happen here? j first of all this was a unintentional mistake. So, this is j.

Now, how will the array be printed here if I do it in this way? What will be printed? Here first thing that we printed is j, a j. What is the value of j? 1. So, what is a j a 1; that is this element. So, 7 will be printed. Suppose I have got a backslash n here. So, 7 will be, 7 will be printed, then 6 will be printed because i is now 2, i is now 3, then 4 will be printed, but till less than 3. So, therefore, 1 2 3 this will not be printed. So, and also again I am missing this one. So, what should I do? What should I do? I can make it j minus 1 what will happen in that case? I am printing j is 1 and I am printing a j minus 1; that means, I am printing a 0 5. Next a j becomes 2. So, I will be printing j minus 1; that means, 2 minus 1 a 1, 7 then 6. So, 7 then j becomes 2 then j becomes 3. So, 3 minus 1 2, I have printed a 2 have printed.

Now, j will be increment it will be 3 the last one will not be printed. So, the here also I have to make it less than equal to 3. Was this one alright? Just quickly check. This is a common source of error I am reading this array. Let me do it again I am reading this array, a 0 then I have read a 0, then i is incremented, so the 4 elements, so I will come up to a 3. So, the second one I am reading the second one a i is 1. So, a 1, i is next 2 a 2 and then a 3. So, this is ok. So, you have to be a little careful about all this.

So, reading and printing, reading and printing we cannot directly do we have also to do it through a loop. So, I cannot do like this.

(Refer Slide Time: 23:13)



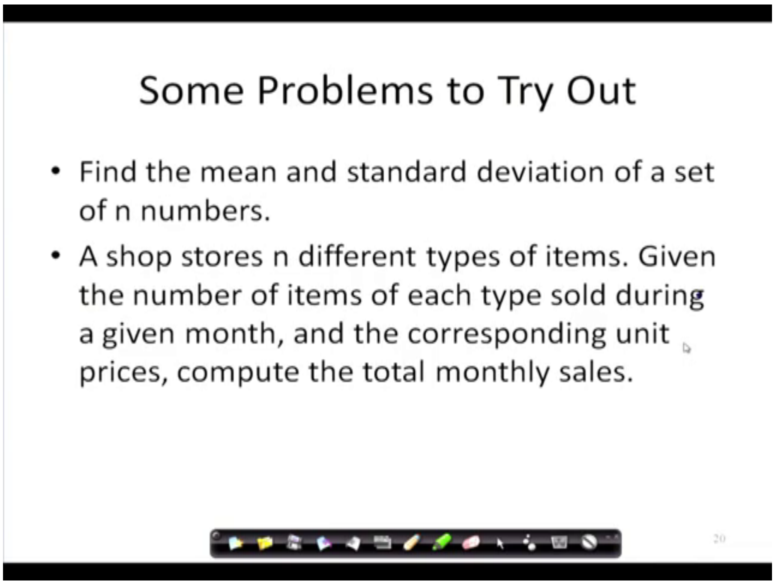
How to copy the elements of one array to another?

- By copying individual elements

```
int a[25],b[25];
 for  (j=0; j<25; j++)
       a[j] = b[j];
```

How to copy elements of one array to another? By copying individual elements I have already shown that. So, that is what we have to say about arrays right. Now, will come to some examples say I want to this is an example of copying an array there are, there are two arrays a 25 and B 25. So, I am trying to copy them. So, you know that I can do it by copying it in this way.

(Refer Slide Time: 23:47)

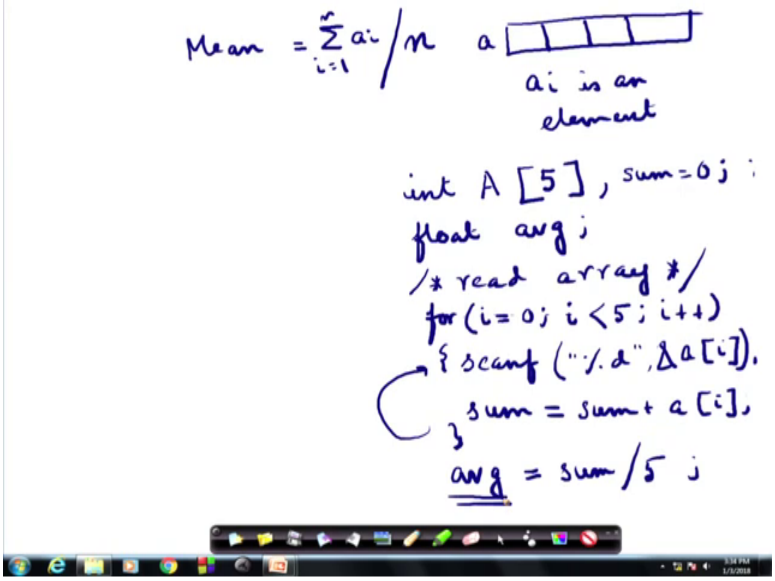


Some Problems to Try Out

- Find the mean and standard deviation of a set of n numbers.
- A shop stores n different types of items. Given the number of items of each type sold during a given month, and the corresponding unit prices, compute the total monthly sales.

Now, some problems we will like to try out. Now, one is finding the mean and standard deviation of a set of n numbers. I leave this standard deviation part we can try to do that mean you know mean is the arithmetic average. So, I will first try to find write a program where all of you will be able to write find the mean and standard deviation of a set of n numbers.

(Refer Slide Time: 24:30)



$$\text{Mean} = \sum_{i=1}^{n} a_i / n$$

```
int A [5], sum=0;
float avg;
/* read array */
for(i=0; i<5; i++)
{ scanf ("%d",&a[i]),
  sum = sum+ a[i],
}
avg = sum/5 ;
```

So, what are you going to do? Mean means the average right. So, I add the numbers all the numbers suppose my array is a. So, each element I designate as a i is an element. So, I will have sigma a i, i varying from 1 to n whatever this size is divided by n that is the average I add all the numbers. So, it is simple to write the program. Suppose I have read the array I have first of all I have to read the elements in an array.

So, summarizing all what we have learnt what should be my first statement first statement is declaring the array. So, let me declare an array, int a suppose the size of the array is 5, maximum size possible. And I also have sum and I have got another variable avg all our integers, again avg may not be integer, mean may not be integer. So, and then I have got another float avg right.

Now, first I have to read the array. So, read the numbers, I am making a shortcut I am leaving out the printf statement please enter the array that part I am leaving out. Then what I am doing here? Here of course, for i assigned 0, i less than suppose there are 5 elements, i less than 5, i plus plus. If it was some other value then I would have put it n here and I had to read the n before I read the array, all right.

So, I do this and then initially I can make it the sum here to be sum is initialized to 0 all right. Now, for i equal to 0, i less than m, i plus plus I read the array I am just now reading the array. So, what I do? Scanf, percentage d, ampersand, a i, I am reading that and then here in this for loop I can repeatedly compute the sum, sum equals sum plus a i. So, I will get the sum here ultimately at the end of the loop, at the end of the loop I will get the sum. So, now, here I can write average is sum divided by whatever the value was I have taken 5 to be a fixed value, so I divide with 5 and that is the average. So, in that way I can find out the mean this value. The next task is computing the standard deviation. So, that I leave to you for the time being. We will take it up later.