

Highest number:

```
#include<stdio.h>

int main(){
    int a[100],i,n;
    scanf("%d",&n);

    for(i=0;i<n;i++){
        scanf("%d",&a[i]);
    }
    for(i=0;i<n;i++){
        if (a[0] < a[i]) {
            a[0] = a[i];
        }
    }
    printf("%d",a[0]);
}
```

Binary Search:

```
#include <stdio.h>

int binarySearch(int [], int, int, int);

int main()
{
    int c, first, last, n, search, array[100], index;
```

```
scanf("%d", &n);  
scanf("%d", &search);
```

```
for (c = 0; c < n; c++)  
    scanf("%d", &array[c]);
```

```
first = 0;  
last = n - 1;
```

```
index = binarySearch(array, first, last, search);
```

```
if (index == -1)  
    printf("false");  
else  
    printf("true");
```

```
return 0;  
}
```

```
int binarySearch(int a[], int s, int e, int f) {  
    int m;
```

```
    if (s > e)
```

```
    return -1;

    m = (s + e)/2;

    if (a[m] == f)
        return m;
    else if (f > a[m])
        return binarySearch(a, m+1, e, f);
    else
        return binarySearch(a, s, m-1, f);
}
```

Summing:

```
#include <stdio.h>

int main()
{
    int t;
    int n;

    while(scanf("%d",&n)==1){
        if(n==0)
            break;

        t=0;
        while(1){

            while(n!=0){
```

```
        t=t+(n%10);
        n=n/10;
    }
    if(t/10==0){
        break;}
    else{
        n=t;
        t=0;
    }
}

printf("%d\n",t);
}

return 0;
}
```

Swap Count:

```
#include<stdio.h>
```

```
int main()
```

```
{  
    int t,n,a[50];  
    scanf("%d",&t);  
    while(t--)  
    {  
        scanf("%d",&n);  
        for(int i=0;i<n;i++)  
        {  
            scanf("%d",&a[i]);  
        }  
        int swap=0;  
        for(int k=0;k<n;k++)  
        {  
            for(int j=0;j<n-1;j++)  
            {  
                if(a[j+1]<a[j])  
                {  
                    int temp=a[j];  
                    a[j]=a[j+1];  
                    a[j+1]=temp;  
                    swap++;  
                }  
            }  
        }  
        printf("Optimal train swapping takes %d swaps.\n",swap);  
    }  
    return 0;  
}
```