

PROGRAMMING FOR PROBLEM SOLVING:

- 4) Write a program to find the maximum temperature in a day (which is stored in an array), in which the temperatures are first increasing order and then decreasing:

* PROGRAM:

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

```
void bubble_sort(int a[], int size)
```

```
{ int i, j, t;
```

```
  for(i=0; i<size-1; i++)
```

```
  { for(j=0; j<size-i-1; j++)
```

```
    { if(a[j]>a[j+1])
```

```
      { t = a[j];
```

```
        a[j] = a[j+1];
```

```
        a[j+1] = t;
```

```
    }
```

```
  }
```

```
}
```

```
}
```

```
void selection_sort(int b[], int size)
```

```
{ int i, j, t, minI;
```

```
  { for(i=0; i<size-1; i++)
```

```
    { minI = i;
```

```
      for(j=i+1; j<size; j++)
```

```
        { if(b[j]>b[minI])
```

```
          minI = j; }
```

```

    t = b[minI];
    b[minI] = b[i];
    b[i] = t;
} }

```

```

}

```

```

int main()

```

```

{
    int i, t;
    int temp[20];
    printf("Enter the number of temperatures to be read:");
    scanf("%d", &t);
    printf("The temperatures in a day are: \n");
    for(i=0; i<t; i++)
    {
        scanf("%d", &temp[i]);
    }
    printf("ASCENDING ORDER: \n");
    bubble_sort(temp, t);
    for(i=0; i<t; i++)
    {
        printf("%d\t", temp[i]);
    }
    printf("\n");
    printf("The maximum temperature is %d\n", temp[t-1]);
    printf("DESCENDING ORDER: \n");
    Selection_sort(temp, t);
    for(i=0; i<t; i++)
    {
        printf("%d\t", temp[i]);
    }
    printf("\n");
    printf("The maximum temperature is %d\n", temp[0]);
    return 0;
}

```

OUTPUT:

Enter the number of temperatures to be read: 7.
The temperatures in a day are:

14
21
35
42
18
10
15.

ASCENDING ORDER:

10 14 15 18 21 35 42.

The maximum temperature is 42.

DESCENDING ORDER:

42 35. 21 18 15. 14 10

The maximum temperature is 42.

- 3) An international standard book number (ISBN) is used to uniquely identify a book. It is made of 10 digits. Write a program using user-defined function that tests an ISBN to see if it is valid. For an ISBN number to be valid, the weighted sum of the 10 digits must be evenly divisible by 11.

* PROGRAM:

```
#include <stdio.h>
void check(int a[])
{
    int i, j, r, count = 0;
    for(i = 0, j = 10; (i < 10) && (j > 0); i++, j--)
    {
        r = (a[i] * j);
        count += r;
    }
    printf("weighted sum = %d\n", count);
}
```



```

if ((count % 11) == 0)
{ printf("Given number is a valid ISBN number"); }
else
{ printf("Given number is not a valid ISBN number"); }
}

int main()
{ int i;
  int ISBN[10];
  printf("Enter the number: \n");
  for(i=0; i<10; i++)
  { scanf("%d", &ISBN[i]); }

  for(i=0; i<10; i++)
  { printf("%d\t", ISBN[i]); }

  printf("\n");
  check(ISBN);
  return 0;
}

```

⇒ OUTPUT:

1) Enter the number:

0
0
7
8
8
1
8
0
9
5

0 0 7 8 8 1 8 0 9 5.

weighted sum = 220

Given number is a valid ISBN number.

2) Enter the number:

1

0

4

5

0

0

2

9

3

5

1

1 0 4 5 0 0 2 9 3 5

Weighted sum = 123

Given number is not a valid ISBN number

2) Write a program for simple guessing game. The computer is to generate a random number between 1 and 20. The user is given upto 5 tries to guess the exact number. After each guess you are to tell the user if the guessed number is greater than, less than or equal to the random number. If it is equal, no more guesses. If the user hasn't guessed the number after 5 tries; display the message that the user should know it by now & terminate the game.

* PROGRAM:

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

```
#include <stdlib.h>
```

```
#include <time.h>
```

```
int main()  
{ int range, num, i, random;  
  srand(NULL);  
  range = ((20-1)+1);  
  random = rand() % range + 1;  
  for(i=1 ; i<=5 ; i++)  
  {  
    printf("Guess a number in the range 1 to 20:");  
    scanf("%d", &num);  
    if(random > num)  
    { printf("%d is less than the random number\n", num); }  
    else if(random < num)  
    { printf("%d is greater than the random number\n", num); }  
    else  
    break;  
  }  
  if(i==6)  
  { printf("%d", random); }  
  else  
  printf("You are the winner");  
  return 0;  
}
```


OUTPUT:

1) Guess a number in the range 1 to 20: 7.
7 is greater than the random number.

Guess a number in the range 1 to 20: 2.
2 is less than the random number.

Guess ~~11~~ a number in the range 1 to 20: 4.
* You are the winner.

2) Guess a number in the range 1 to 20: 18.
18 is greater than the random number.

Guess a number in the range 1 to 20: 16.
16 is greater than the random number.

Guess a number in the range 1 to 20: 13.
13 is greater than the random number.

Guess a number in the range 1 to 20: 10.
10 is greater than the random number.

Guess a number in the range 1 to 20: 8.
8 is greater than the random number.

Random number = 4.