Sub: ESFP – I
Course Code: 2CSE102
Practical – 7

Name:Dwij vatsal desai
Roll No:BDA-08
Branch:BDA
Class: B
Batch:14

Q.1.Problem Definition:

The policy followed by a company to process customer orders is given by the following
rules:
1. If a customer order quantity less than or equal to that in stock and his credit is OK,
supply his requirement.
2. If his credit is not OK do not supply. Send him an intimation.
3. If his credit is OK but the item in stock is less than his order quantity, supply what is in
stock. Intimate to him that the balance will be shipped.
4. Assume, you have 100 quantities of computer available in stock.

Algorithm:

Step 1: Start
Step 2: Declare the variable order, credit and constant variable stock with predefined value 100.
Step 3: Take the values of order and credit.

Step 4: Using the if else ladder and using the following formula find the , order > stock derived results. order <= stock && (credit == '1'), credit == '0'.
Step 5: Display the results.
Step 6: Display the remaining stock.
Step 7: End.

Solution:

Code:-

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

void main()

{

    int stock = 100;
    int order;
    char credit;

    printf("Enter the order quantity: ");
    scanf("%d",&order);

    printf("Enter the credit (1 for OK, 0 for not OK): ");
    scanf(" %c",&credit);

    if (order <= stock && (credit == '1'))
    {

        printf("Order will be supplied.\n");
        stock -= order;
```

```c
    }

    else if (credit == '0')
    {

        printf("Credit is not OK. Order will not supplied. Intimation
sent.\n");

    }


    else if (order > stock)
    {

        printf("Partial supply: %d computers supplied.\n", stock);
        stock = 0;

    }

    printf("Remaining stock: %d\n", stock);

}
```

```
Enter the order quantity: 34
Enter the credit (1 for OK, 0 for not OK): 1
Order will be supplied.
Remaining stock: 66


...Program finished with exit code 0
Press ENTER to exit console.
```

```
Enter the order quantity: 34
Enter the credit (1 for OK, 0 for not OK): 0
Credit is not OK. Order will not supplied. Intimation sent.
Remaining stock: 100


...Program finished with exit code 0
Press ENTER to exit console.
```

Q.2.Problem Definition:

A certain grade of steel is graded according to the following
conditions:
1. Hardness must be greater than 50
2. Carbon content must be less than 0.7
3. Tensile strength must be greater than 5600
The grade are as follows:
Grade is 10 if all three conditions are met.
Grade is 9 if condition (i) and (ii) are met.
Grade is 8 if condition (ii) and (iii) are met.
Grade is 7 if condition (i) and (iii) are met.
Grade is 6 if only one condition is met.
Grade is 5 if none of the conditions are met.
Make a proper program, which will require the user to give value of
hardness, carbon
content and tensile strength of the steel under consideration and as
per the given input,
output the grade of the steel.

Algorithm:
Step 1: Start
Step 2: Declare the variables hard, tensile, car and constant variables
hardness,
tensile strength and carbon.

Step 3: Take the values of car hardness and tensile from the user and read them.
Step 4: By using the if else ladder find the derived result.
Step 5: Display the result found.
Step 6: End.

Solution:

<u>Code:-</u>

```c
#include <stdio.h>
void main()
{

    const int hardness = 50, tensile_strength = 5600;  const float carbon = 0.7;

    int h, t;              float c;
    //h=hardness t=tensile     //c=carbon
    printf("Enter the hardness and the tensile strength of the steel: ");
    scanf("%d%d", &h, &t);

    printf("Enter the carbon content in that steel: ");
    scanf("%f", &c);

    if (h >= hardness && t >= tensile_strength && c <= carbon)
    {

        printf("The grade of the steel is 10.\n");

    }
```

```c
else if (h >= hardness && c <= carbon && t < tensile_strength)
{

    printf("The grade of the steel is 9.\n");

}

else if (c >= carbon && t >= tensile_strength && h < hardness)
{

    printf("The grade of the steel is 8.\n");

}

else if (h >= hardness && t >= tensile_strength && c != carbon)
{

    printf("The grade of the steel is 7.\n");

}

else if (h >= hardness || c <= carbon || t >= tensile_strength)
{

    printf("The grade of the steel is 6.\n");

}

else if (h < hardness && c > carbon && t < tensile_strength)
{

    printf("The grade of the steel is 5.\n");
```

}

}

```
Enter the hardness and the tensile strength of the steel: 53 5600
Enter the carbon content in that steel: 0.5
The grade of the steel is 10.


...Program finished with exit code 0
Press ENTER to exit console.
```

Q.3.Problem Definition:
Generate the following series with the help of goto keyword.
1. Generate series of natural number.
1, 2, 3, 4, 5, 6, 7, 8, 9, 10, ...
2. Generate table of any random number.
5, 10, 15, 20, 25, 30, 35, 40, 45, 50
3. Generate series like:
6,11,16,21,26,.....

Algorithm:
Step 1: Start

Step 2: Declare the constant variables n=1, num=5, table=5.
Step 3: Use the goto function and make the required results.
Step 4: Display the answers.
Step 5: End.

Solution:
            Code:-
#include <stdio.h>

```c
void main()
{
    int n = 1;  int num = 5;  int table = 5;

    printf("The first series is natural numbers as follows:\n\n");

    repeat1:

    if (n <= 10)
    {
        printf("%d,",n);

        n++;

        goto repeat1;
    }

    printf("\n\nThe second series is of table of %d as follows:\n\n", num);

    repeat2:

    if (num <= 50)
    {

        printf("%d,", num);

        num += table;

        goto repeat2;

    }
```

```c
    printf("\n\nThe third series is general series as follows:\n\n");

    int custom = 6;

    repeat3:

    if (custom <= 26)
    {
        printf("%d,", custom);

        custom += 5;

        goto repeat3;

    }

}
```

```
The first series is natural numbers as follows:

1,2,3,4,5,6,7,8,9,10,

The second series is of table of 5 as follows:

5,10,15,20,25,30,35,40,45,50,

The third series is general series as follows:

6,11,16,21,26,

...Program finished with exit code 3
Press ENTER to exit console.
```