Rami Wail Shoula: Day 1 Tasks

I- Print "Hello World

```
170        printf("Q1: \nHello World!\n");
171        //std::cout << "Hello World!\n"; //cpp equivalent
```

```
Q1:
Hello World!
```

II- Print ASCII code user-input character

```
173        //Q2
174        char c; //initialize character
175        printf("Q2: \nPlease enter a character: "); //req. user input
176        scanf("%c", &c); //user input
177        //std::cin >> c; //cpp equivalent
178        get_ASCII_value(c); //calling ascii conversion function
```

```
16    //func. that returns text user specific (void) for Q2
17    void get_ASCII_value(char c)
18    {
19        printf("The ASCII value of %c = %d\n", c, c); //prints character and ascii equivalent no.
20    }
```

```
Q2:
Please enter a character: q
The ASCII value of q = 113
```

III- Implement Magic box algorithm on numbers from 1-9, so that numbers are displayed , sum of any row equals sum of any column equals 15 .

```
180        //Q3 magic box 3x3 sum=15
181        int n = 3; // Works only when n is odd
182        printf("Q3: magic box 3x3 sum=15\n");
183        generateSquare(n);
```

```
Q3: magic box 3x3 sum=15
The Magic Square for n=3:
Sum of each row or column 15:

2 7 6
9 5 1
4 3 8
```

IV- -Implement Magic box for odd box of order n

```
185    //Q4
186    // Works only when n is odd
187    printf("Q4: Please enter an odd number for magic box generation: "); //req. user input
188    scanf("%d", &n); //user input
189    // checking if no. is odd, true if num is not perfectly divisible by 2
190    while(n % 2 == 0 || n<0 ) //if remainder when divided by 2 is 0 then it is even, request odd input
191    {
192        printf("%d is even. or negative", n);
193        printf("\nkidly re-enter an 'odd' +ve number for magic box generation: ");
194        scanf("%d", &n); //user input
195    }
196    generateSquare(n);
```

```
22    //func. that generates odd sized magic squares for Q3,4
23    void generateSquare(int n)
24    {
25        //The magic constant of a normal magic square (odd n)
26        //M = n(n^2+1)/2
27        //ex: n=3 then M = 3*(3^2+1)/2 = 3*(9+1)/2 = 3*5 = 15
28
29        int magicSquare[n][n]; //initialize 2D array (matrix) for magic square //works on any c compiler other than visual studio (legacy) compilers
30        memset(magicSquare, 0, sizeof(magicSquare)); //set all entries in matrix to zeroes using built in func.'s memset and sizeof
31        //position to place the initial 1 (relative to n size)
32        int i = n / 2; //x loc (row)
33        int j = n - 1; //y loc (column)
34        // Use 3 conditions to individually place values in magic square //one for loop
35        for (int loop = 1; loop <= n * n;)   //one loop over stretched entire 2D plane in one dimension for simplicity
36        {
37            if (i == -1 && j == n) // 3rd condition //resetting
38            {
39                j = n - 2;
40                i = 0;
41            }
42            else {
43                // 1st condition helper if next number exceeds square's right side // resetting columns
44                if (j == n)
45                    j = 0;
46                // 1st condition helper if next number exceeds square's top side // resetting rows
47                if (i < 0)
48                    i = n - 1;
49            }
50            if (magicSquare[i][j]) // 2nd condition //resetting
51            {
52                j -= 2;    //resettig columns
53                i++;
54                continue; //increment row and continue
55            }
56            else
57                magicSquare[i][j] = loop++; // set loop increment no.
58
59            j++;
60            i--; // 1st condition
61        }
```

```
62        // Print magic square
63        printf("The Magic Square for n=%d:\nSum of "
64            "each row or column %d:\n\n",
65            n, n * (n * n + 1) / 2);
66        for (i = 0; i < n; i++) {
67            for (j = 0; j < n; j++)
68                printf("%d ", magicSquare[i][j]); //"%3d "
69            printf("\n");
70        }
71    }
```

```
Q4: Please enter an odd number for magic box generation: 7
The Magic Square for n=7:
Sum of each row or column 175:

20 12 4 45 37 29 28
11 3 44 36 35 27 19
2 43 42 34 26 18 10
49 41 33 25 17 9 1
40 32 24 16 8 7 48
31 23 15 14 6 47 39
22 21 13 5 46 38 30
Press any key to continue . . .
```

V-   Write a console program to display a switch menu of 5 options

```
198         //Q5
199         system("pause"); //pauses screen awaiting user input
200         system("cls");   //clear screen
201         quickmenu(); //call meny function
202
203         return 0;
204  }
```

```
73   void quickmenu() //menu function for Q5
74 ⊟ {
75       //string menu[5]
76 ⊟     char menu[5][100] = { "Calculate the Sum and the Average of an array of integers",
77       "Print Hello World",
78       "Find the tallest and the shortest sentences of the user-input sentences",
79       "Find the Treasure Game",
80       "Exit" };
81
82       int x = 0;
83
84 ⊟     while (true) {
85           system("cls"); //clears screen
86
87           SetConsoleTextAttribute(GetStdHandle(STD_OUTPUT_HANDLE), 14);
88           printf( "Q5: Main menu: \n\n");
89
90 ⊟         for (int i = 0; i < 5; i++) {
91               if (i == x)
92 ⊟             {
93                   SetConsoleTextAttribute(GetStdHandle(STD_OUTPUT_HANDLE), 2);
94                   //menu[i] ="ABC";
95                   printf("%s\n",menu[i]); //finally got it to work!!! yay!!!
96                   //cout << menu[i] << endl; //i used cout here bec. i was unable to print a string with printf function sadly
97               }
98 ⊟             else {
99                   SetConsoleTextAttribute(GetStdHandle(STD_OUTPUT_HANDLE), 14);
100                  //cout << menu[i] << endl;
101                  printf("%s\n",menu[i]);
102              }
103          }
```

```c
            while (true) {
                if (GetAsyncKeyState(VK_ESCAPE)!= 0)
                {
                    break;
                }
                if (GetAsyncKeyState(VK_UP) != 0) {
                    x = x - 1;
                    if (x == -1) {
                        x = 4;
                    }
                    break;
                }

                else if (GetAsyncKeyState(VK_DOWN) != 0) {
                    x = x + 1;
                    if (x == 5) {
                        x = 0;
                    }
                    break;
                }
                else if (GetAsyncKeyState(VK_HOME) != 0) {
                    x = 0;
                    break;
                }

                else if (GetAsyncKeyState(VK_END) != 0) {
                    x = 4;
                    break;
                }

            else if (GetAsyncKeyState(VK_RETURN) != 0) { //enter key
                //system("cls");
                switch (x) // switch cases for menu
                {
                case 0: printf( "\nYou selected: Calculate the Sum and the Average of an array of integers.\n");
                    break;
                case 1: printf( "\nYou selected: Hello world\n");
                    break;
                case 2: printf( "\nYou selected: Find the tallest and the shortest sentences of the user-input sentences\n");
                    break;
                case 3: printf( "\nYou selected: Find the Treasure Game\n");
                    break;
                case 4: printf( "\nYou selected: Exit, Program is shutting down now.\n");
                    break;
                }
                system("pause"); //pauses screen awaiting user input
            }

        }
    }
}
```

```
Q5: Main menu:

Calculate the Sum and the Average of an array of integers
Print Hello World
Find the tallest and the shortest sentences of the user-input sentences
Find the Treasure Game
Exit

You selected: Calculate the Sum and the Average of an array of integers.
Press any key to continue . . .
```

Here 'Home' and 'End' keys take menu to top and bottom, respectively.

'Up' and 'Down' arrow keys take menu up and down, respectively and in extreme cases resets menu to the Bottom/Top, respectively.

```
Q5: Main menu:

Calculate the Sum and the Average of an array of integers
Print Hello World
Find the tallest and the shortest sentences of the user-input sentences
Find the Treasure Game
Exit

You selected: Exit, Program is shutting down now.
Press any key to continue . . .
```

Also, 'Esc' key exits the menu to the previous selection screen (clears selection).

I uploaded the entire file as a devc++ project (.dev).
kindly run the code on devc++ as I had issues with visual studio installation.
(specifically the only issue from visual studio is that the 2D array as not accepted with variable size on line 29: int magicSquare[n][n]; , this works on online compilers and on devc and codeblocks)

Also, I changed the menu from an array of strings to a 2D array of characters so I can use printf instead of cout. And reuploaded the files (now fully C based only)

VI-   Entire code run example:

```
Q1:
Hello World!
Q2:
Please enter a character: 1
The ASCII value of 1 = 49
Q3: magic box 3x3 sum=15
The Magic Square for n=3:
Sum of each row or column 15:

2 7 6
9 5 1
4 3 8
Q4: Please enter an odd number for magic box generation: 5
The Magic Square for n=5:
Sum of each row or column 65:

9 3 22 16 15
2 21 20 14 8
25 19 13 7 1
18 12 6 5 24
11 10 4 23 17
Press any key to continue . . .
```

```
Q5: Main menu:

Calculate the Sum and the Average of an array of integers
Print Hello World
Find the tallest and the shortest sentences of the user-input sentences
Find the Treasure Game
Exit

You selected: Find the tallest and the shortest sentences of the user-input sentences
Press any key to continue . . .
```