Prepared by: *Mohamed Abdulqader*
Senior Software Engineer

C Assignments

Lesson 02-Operators & Decision making – if statement

Task_id_001: (lesson 01-Intro to C-Programming)
Write a program that displays a simple menu with 3 choices
a) Good morning.
b) Good evening.
c) Exit Program.

* Asks the user to enter the required choice ‘a’, ‘b’ or ‘c’.
* When the user presses ‘a’, ‘b’ or ‘c’ (without pressing enter key), then the program will delete the whole menu, display the corresponding entry (ex. if user entered ‘a’, display “Good morning” or display “exit the program” if pressed ‘c’), after displaying the option wait for the user to press any key, close the program when any key is pressed.
* If wrong input is entered (Not ‘a’, ‘b’ or ‘c’), the program clears the screen then prints "Wrong choice!", then it closes.
Hint: For this task you may need: system("cls") when using codeblocks.

Lesson 03-Decision Making - Continue

Task_id_002:
Rewrite the previous task "Task_id_001" using the new statements (i.e. operators, if conditions) introduced in lesson “03-Decision Making“

Lesson 04-Loops

Task_id_003:
Write a program using the menu example shown at the end of current lesson: Edit the program for the following:
1- Support capital and small letters (Example: ‘a’ or ‘A’ to display Good morning) and so on.
2- Add a new menu entry to let the user exit the program when ‘E’ or ‘e’ key is pressed.
Use while loop instead of do-while and use switch cases instead of if statement.

Task_id_004:
- Write a program to create a function accepting one input parameter 'int' and print it in reversed order, number should be 5 may be 5 digits or less.
Example: input: 4567 > output printed 7654

Prepared by: *Mohamed Abdulqader*
Senior Software Engineer

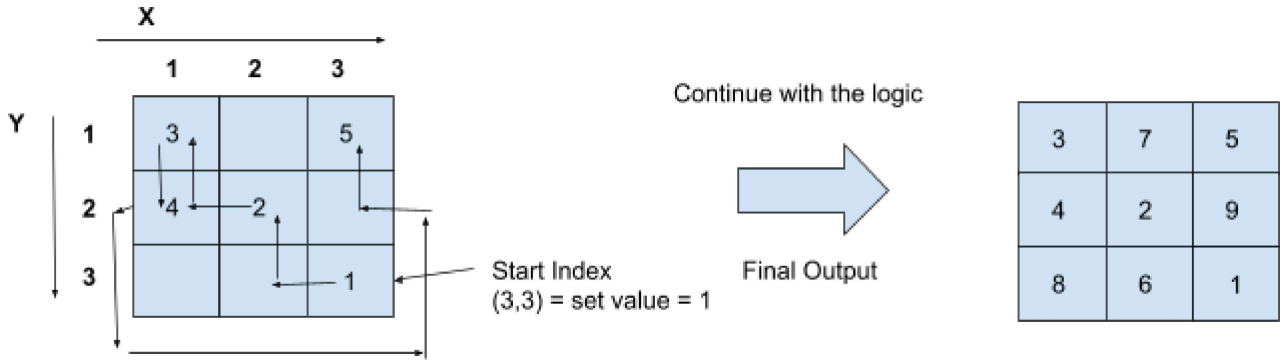
Lessons 05-Arrays, 06-Functions

Task_id_005:
- Write a program to create a function (Implementation in c & prototype in h files) that accepts 2 input parameters 'int Arr[ ], int arrSize' and print maximum & minimum numbers inside the array.

Task_id_006:
Write a program that continues asking about an integer number and accumulate the input integers and when the sum is > 100, it should print this sum and exit the program.

(Bonus - Puzzle, not related to arrays topic!)

Task_id_007:
Magic box, n*n
The program should ask about the n value and the start index (x,y)
hint : it is recommended to restrict the n value to max value , to match your console size!
Example: let's assume n = 3, and the start index is 3,3

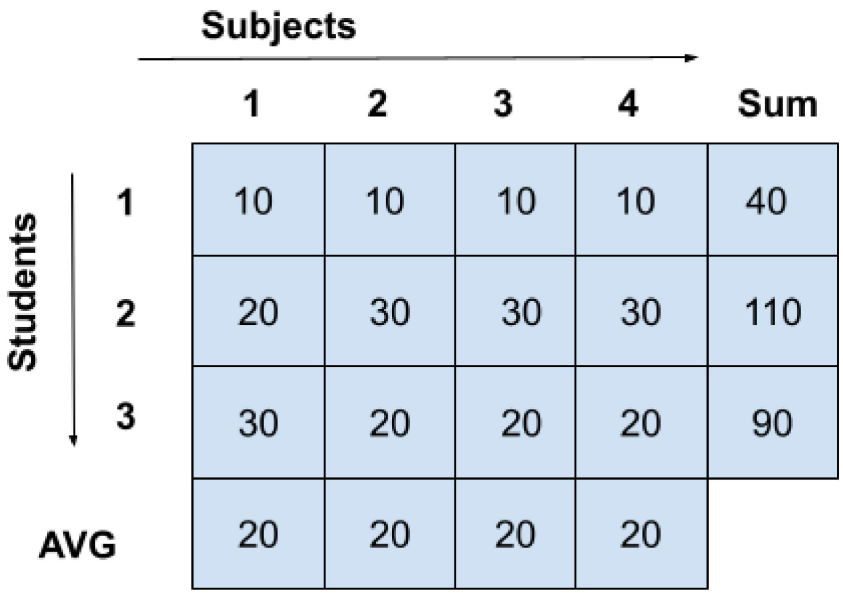


Logic:
- Put 1 at the start index
- If current value (ex: 1, 2, 4, 5, 7, 8) is not divisible by n (3): Move left + up
- If value is divisible by n (ex: 3, 6, 9): move down

Hint: if reached borders >> round to the other side.
Funny idea of the magic box:
that if you applied the same logic to the last value, it will go with you to the start index

Task_id_008:
Assume you have 3 students having exam results for 4 subjects
Create 2-D array to collect this data from the user
Then print the sum of every student exam grades and the average of degrees per subject as well as printing the 2-D matrix in a grid showing the sum as an additional column and the avg as an additional row.

Ex for output:



Task_id_009:
Get a key from the keyboard from the user and then display its ASCII code for either extended or non extended keys.
Hint: search google about the extended and non extended keys and what are their ASCII code value
Hint: use getch() from conio.h to get user keystrokes
External Note for ANSI C 89 compilers (not applicable for gcc or codeblocks): for extended keys, the first char is equal to null **'\0'** and you need to get the second char for value.

Task_id_010:
get a string from user char by char and assume this string is completed either if the user pressed enter or number of char exceeds a maximum value 40
Then display the whole string.
Hint: use getch() from conio.h to get user keystrokes

Task_id_011:
Use prev assign to get the first and last name from the user and display the full name.

Task_id_012:
Create highlight menu and display it

New_

Display

Exit

Hint: use getch() from conio.h to get user keystrokes
And allow user to use up and down, home, end, tab, esc and enter keyboard keys to navigate this menu
Hint: highlighted line is indicated by ending the line by underscore "_"
Down will highlight the next line, if reached the last, go to the first one again.
Up key will do the vice versa.
Home: will go to the top of the menu
End: will go to the bottom of the menu
Esc: exit the program
Tab: go next one (same as down)
Enter: depends on the current highlighted line:
    if it is New, print "new", then exit the program if user entered anykey
    If it is Display, print "display", then exit the program if user entered anykey
    If it is Exit, exit the program