week 2_3:

patterns

rajin teaches programming

INTRODUCTION:

In programming, we can create various interesting patterns using loops. Although hard to understand why this is necessary at all first, pattern printing helps you really understand the crux of looping structures and how they work. This becomes especially useful in data science when we have to access certain elements in a matrix for calculating things like determinants, and so on and so forth. Thus, this is an essential skill to master. These patterns can be achieved by controlling the iterations and conditions within loops. Let's start with some simple patterns and gradually move on to more complex ones.

- PRINTING PATTERNS USING FOR LOOPS:

You already remember the basic structure used in for-loops. We will mostly be using this structure to print different patterns by manipulating the loop parameters.

code for printing a simple square,

Developer - square.c

```c
#include<stdio.h>

int main() {

    int height = 5;

    for (int i=1; i<=height; i++) {

        for (int j=1; j<=height; j++) {

            printf("*");
        }
        printf("\n");
    }

    return 0;
}
```

```
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
```

along with the output.

As you may understand, providing simple methods on how to do these kinds of problems defeats the problem solving aspect of programming. Thus, I will be presenting you with problems in this worksheet, and we will work them out throughout the duration of the class. It is integral for your learning that you do not look up solutions online until you absolutely feel like you cannot solve it. All of this is to improve your problem solving skills, and hone your sharpness as a programmer.
All that being said, here are the questions:

– Print a rectangular pattern of size m*n, where m and n are dimensions input by the user.

– Print a hollow box pattern of n lines, where n is a dimension input by the user.

– Print a hollow box pattern of size m*n, where m and n are dimensions input by the user.

– Print a hollow rhombus of n lines, where n is a dimension input by the user.

– Print a hollow parallelogram of size m*n, where m and n are dimensions input by the user.

– Print a left-aligned right-angled triangle for n number of lines, where n is a dimension entered by the user.

– Print a left-aligned, hollow, right-angled triangle for n number of lines, where n is a dimension entered by the user.

– Print a right-aligned right-angled triangle for n number of lines, where n is a dimension entered by the user.

– Print a right-aligned, hollow, right-angled triangle for n number of lines, where n is a dimension entered by the user.

next class 3_1:

functions

rajin teaches programming

reach out for classes at: rajin.khan2001@gmail.com