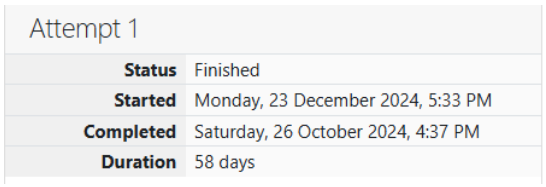
Week 3-2

Operators and Expressions, Managing Input and Output Operations

Roll no: 240801162

Name: Keerthanaa Kumaraswamy



Problem 1: Write a program that determines the name of a shape from its number of sides. Read the number of sides from the user and then report the appropriate name as part of a meaningful message. Your program should support shapes with anywhere from 3 up to (and including) 10 sides. If a number of sides outside of this range is entered then your program should display an appropriate error message.

Sample Input 1

3

Sample Output 1

Triangle

Sample Input 2

7

Sample Output 2

Heptagon

Sample Input 3

11

Sample Output 3

The number of sides is not supported.

Code:

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

    int n;
    scanf("%d",&n);


          if (n == 3){
            printf("Triangle");
          }


           else if ( n == 4){
            printf("Square");
           }


          else if  ( n == 5){
            printf("Pentagon");
          }


          else if  ( n == 6){
            printf("Hexagon");
          }


         else if  ( n == 7 ){
            printf("Heptagon");
         }


          else if  (  n == 8 ){
            printf("Octagon");
          }

         else if  (   n == 9 ){
            printf("Nonagon");
         }


           else if  ( n == 10){
            printf("Decagon");
           }

      else{
            printf("The number of sides is not supported.");

    }
    return 0;
```

OUTPUT:

	Input	Expected	Got	
✓	3	Triangle	Triangle	✓
✓	7	Heptagon	Heptagon	✓
✓	11	The number of sides is not supported.	The number of sides is not supported.	✓

Passed all tests! ✓

Problem 2: The Chinese zodiac assigns animals to years in a 12-year cycle. One 12-year cycle is

shown in the table below. The pattern repeats from there, with 2012 being another year of the Dragon, and 1999 being another year of the Hare.

Year	Animal
2000	Dragon
2001	Snake
2002	Horse
2003	Sheep
2004	Monkey
2005	Rooster
2006	Dog
2007	Pig
2008	Rat
2009	Ox
2010	Tiger
2011	Hare

Write a program that reads a year from the user and displays the animal associated with that year. Your program should work correctly for any year greater than or equal to zero, not just the ones listed in the table.

Sample Input 1

2004

Sample Output 1

Monkey

Sample Input 2

2010

Sample Output 2

Tiger

Explanation:

 The program reads the year input from the user.

 It calculates the index by using the formula (year - 2000) % 12.

o Here, 2000 is chosen as the reference year because it corresponds to "Dragon."

o The % 12 operation ensures the cycle repeats every 12 years.

o If index is negative (when the year is before 2000), it adjusts by adding 12.

 The animals array holds the animal names in order, so the calculated index directly

points to the correct animal.

Code :

```c
#include <stdio.h>
int main()
{
    int year;
    const char*animals[] = {"Dragon","Snake","Horse","Sheep","Monkey","Rooster","Dog", :","Pig","Rat","Ox","Tiger","Hare"};
    scanf("%d", &year);
    int index = (year-2000)%12;
    if(index < 0){
        index += 12;
    }
    printf("%s\n",animals[index]);
```

OUTPUT:

	Input	Expected	Got	
✓	2004	Monkey	Monkey	✓
✓	2010	Tiger	Tiger	✓

Passed all tests! ✓

Problem 3: Positions on a chess board are identified by a letter and a number. The letter identifies

the column, while the number identifies the row, as shown below:

Write a program that reads a position from the user. Use an if statement to determine if the column begins with a black square or a white square. Then use modular arithmetic to report the color of the square in that row. For example, if the user enters a1 then your program should report that the square is black. If the user enters d5 then your program should report that the square is white. Your program may assume that a valid position will always be entered. It does not need to perform any error checking.

Sample Input 1

a 1

Sample Output 1

The square is black.

Sample Input 2

d 5

Sample Output 2

The square is white.

Explanation of Changes:

 Now, if (columnIndex + row) % 2 == 0, the program prints "The square is white."

 If (columnIndex + row) % 2 == 1, it prints "The square is black."

Code:

```c
#include <stdio.h>
int main()
{
    char column;
    int row;
    scanf("%c%d",&column,&row);
    int columnindex = column - 'a';
    if((columnindex + row)%2 == 0){
        printf("The square is white.\n");
    }
    else{
        printf("The square is black.\n");
    }
    return 0;
}
```

OUTPUT:

	Input	Expected	Got	
✓	a 1	The square is black.	The square is black.	✓
✓	d 5	The square is white.	The square is white.	✓

Passed all tests! ✓