**WEEK 1 1.1**

1. This is a simple challenge to help you practice printing to stdout.|  
     
   We're starting out by printing the most famous computing phrase of all time! In the editor below, use either printf or cout to print the string *Hello, World!* to stdout.

Input Format  
  
You do not need to read any input in this challenge.  
  
Output Format  
  
Print *Hello, World!* to stdout.  
  
Sample Output  
  
Hello, World!

#include<stdio.h>

int main() {

printf("Hello, World!");

return 0;

}

|  | **Expected** | **Got** |  |
| --- | --- | --- | --- |
|  | Hello, World! | Hello, World! |  |

|  |
| --- |
|  |

2.This challenge will help you to learn how to take a character, a string and a sentence as input in C.

To take a single character ***ch*** as input, you can use scanf("%c", &ch); and printf("%c", ch) writes a character specified by the argument char to stdout:

char ch;

scanf("%c", &ch);

printf("%c", ch);

This piece of code prints the character ***ch***.

#include<stdio.h>

int main() {

char ch;

scanf("%c",&ch);

printf("%c", ch);

return 0;

| **Input** | **Expected** | **Got** |  |
| --- | --- | --- | --- |
| C | C | C |  |  |

3.The fundamental data types in c are int, float and char. Today, we're discussing int and float data types.  
  
The printf() function prints the given statement to the console. The syntax is printf("format string",argument\_list);. In the function, if we are using an integer, character, string or float as argument, then in the format string we have to write %d (integer), %c (character), %s (string), %f (float) respectively.  
  
The scanf() function reads the input data from the console. The syntax is scanf("format string",argument\_list);. For ex: The scanf("%d",&number) statement reads integer number from the console and stores the given value in variable ***number***.  
  
To input two integers separated by a space on a single line, the command is scanf("%d %d", &n, &m), where ***n*** and ***m*** are the two integers.

#include<stdio.h>

int main() {

int n1,n2 ;

float x,y;

scanf("%d %d\n",&n1,&n2);

scanf("%f%f\n",&x,&y);

printf("%d %d\n", n1+n2,n1-n2);

printf("%.1f %.1f\n", x+y,x-y);

return 0;

}

|  | **Input** | **Expected** | **Got** |
| --- | --- | --- | --- |
|  | 10 4  4.0 2.0 | 14 6  6.0 2.0 | 14 6  6.0 2.0 |  |
|  | 20 8  8.0 4.0 | 28 12  12.0 4.0 | 28 12  12.0 4.0 |  |

**WEEk 1 1.2**

1.Write a program to input a name (as a single character) and marks of three tests as m1, m2, and m3 of a student considering all the three marks have been given in integer format.  
  
Now, you need to calculate the average of the given marks and print it along with the name as mentioned in the output format section.  
  
All the test marks are in integers and hence calculate the average in integer as well. That is, you need to print the integer part of the average only and neglect the decimal part.

#include<stdio.h>

int main () {

char name;

int m1,m2,m3;

int avg;

scanf("%c",&name);

scanf("%d %d %d", &m1,&m2,&m3);

avg=(m1+m2+m3)/3;

printf("%c\n", name);

printf("%d", avg);

return 0;

}

|  | **Input** | **Expected** | **Got** |
| --- | --- | --- | --- |
|  | A  3 4 6 | A  4 | A  4 |  |
|  | T  7 3 8 | T  6 | T  6 |  |
|  | R  0 100 99 | R  66 | R  66 |  |

2.Some *C*data types, their format specifiers, and their most common bit widths are as follows:

· *Int ("%d"):* 32 Bit integer  
· *Long ("%ld"):* 64 bit integer  
· *Char ("%c"):* Character type  
· *Float ("%f"):* 32 bit real value  
· *Double ("%lf"):* 64 bit real value

#include<stdio.h>

int main () {

int i;

long ld;

char c;

float f;

double lf;

scanf(" %d %ld %c %f %lf ",&i,&ld,&c,&f,&lf);

printf("%d\n",i);

printf("%ld\n",ld);

printf("%c\n",c);

printf("%.3f\n",f);

printf("%.9lf",lf);

return 0;

}

|  | **Input** | **Expected** | **Got** |
| --- | --- | --- | --- |
|  | 3 12345678912345 a 334.23 14049.30493 | 3  12345678912345  a  334.230  14049.304930000 | 3  12345678912345  a  334.230  14049.304930000 |  |

3.Write a program to print the ASCII value and the two adjacent characters of the given character.  
  
Input  
  
E  
  
Output  
  
69  
D F

#include<stdio.h>

int main () {

char ch;

scanf("%c",&ch);

printf("%d\n",ch);

printf("%c %c\n",ch-1,ch+1);

return 0;

}

|  | **Input** | **Expected** | **Got** |
| --- | --- | --- | --- |
|  | E | 69  D F | 69  D F |  |

**WEEk 2 2.1**

1.Many people think about their height in feet and inches, even in some countries that primarily use the metric system. Write a program that reads a number of feet from the user, followed by a number of inches. Once these values are read, your program should compute and display the equivalent number of centimeters.

Hint:

One foot is 12 inches.

One inch is 2.54 centimeters.

#include<stdio.h>

int main() {

int feet,inches;

float cm;

scanf("%d", &feet);

scanf("%d",&inches);

cm=(feet\*12+inches)\*2.54;

printf("%.2f\n",cm);

return 0;

}

| **Input** | **Expected** | **Got** |  |
| --- | --- | --- | --- |
|  | 5  6 | 167.64 | 167.64 |  |

2.Create a program that reads two integers, a and b, from the user. Your program should compute and display: • The sum of a and b • The difference when b is subtracted from a • The product of a and b • The quotient when a is divided by b • The remainder when a is divided by b

Input Format

First line, read the first number.

Second line, read the second number.

Output Format

First line, print the sum of a and b

Second line, print the difference when b is subtracted from a

Third line, print the product of a and b

Fourth line, print the quotient when a is divided by b

Fifth line, print the remainder when a is divided by b

Sample

Input 1 100 6

Sample Output

106 94 600 16 4

#include<stdio.h>

int main() {

int a,b;

scanf("%d", &a);

scanf("%d",&b);

printf("%d\n%d\n%d\n%d\n%d\n", a+b,a-b,a\*b,a/b,a%b);

return 0;

}

|  | **Input** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- |
|  | 100  6 | 106  94  600  16  4 | 106  94  600  16  4 |  |

3.Question text

A bakery sells loaves of bread for $3.49 each. Day old bread is discounted by 60 percent. Write a program that begins by reading the number of loaves of day old bread being purchased from the user. Then your program should display the regular price for the bread, the discount because it is a day old, and the total price. Each of these amounts should be displayed on its own line with an appropriate label. All of the values should be displayed using two decimal places.  
  
Input Format  
  
Read the number of day old loaves.  
  
Output Format  
  
First line, print Regular price: price  
Second line, print Discount: discount  
Third line, print Total: total  
  
  
Sample Input 1  
  
10  
  
Sample Output 1  
  
Regular price: 34.90  
Discount: 20.94  
Total: 13.96

#include<stdio.h>

int main() {

int loaves;

float regularprice,discount,total;

scanf("%d", &loaves);

regularprice=loaves\*3.49;

discount=regularprice\*0.60;

total=regularprice-discount;

printf("Regular price: %.2f\n",regularprice);

printf("Discount: %.2f\n",discount);

printf("Total: %.2f\n",total);

return 0;

}

|  | **Input** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- |
|  | 10 | Regular price: 34.90  Discount: 20.94  Total: 13.96 | Regular price: 34.90  Discount: 20.94  Total: 13.96 |  |

**WEEK2.2**

1.Top of Form

Goki recently had a breakup, so he wants to have some more friends in his life. Goki has N people who he can be friends with, so he decides to choose among them according to their skills set Yi(1<=i<=n). He wants atleast X skills in his friends. Help Goki find his friends.

#include<stdio.h>

int main() {

int x,y;

scanf("%d", &x);

scanf("%d", &y);

if(y>=x) {

printf("YES\n");

}

else {

printf("NO\n");

}

return 0;

}

Top of Form

|  | **Input** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- |
|  | 100  110 | YES | YES |  |
|  | 100  90 | NO | NO |  |

Question text

Before the outbreak of corona virus to the world, a meeting happened in a room in Wuhan. A person who attended that meeting had COVID-19 and no one in the room knew about it! So everyone started shaking hands with everyone else in the room as a gesture of respect and after meeting unfortunately everyone got infected! Given the fact that any two persons shake hand exactly once, Can you tell the total count of handshakes happened in that meeting?  
  
Say no to shakehands. Regularly wash your hands. Stay Safe.

#include<stdio.h>

int main() {

long long N;

scanf("%lld", &N);

long long handshakes;

handshakes=N\*(N-1)/2;

printf("%lld\n",handshakes);

return 0;

}

|  | **Input** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- |
|  | 1 | 0 | 0 |  |
|  | 2 | 1 | 1 |  |

Question text

3.In our school days, all of us have enjoyed the Games period. Raghav loves to play cricket and is Captain of his team. He always wanted to win all cricket matches. But only one last Games period is left in school now. After that he will pass out from school.  
  
So, this match is very important to him. He does not want to lose it. So he has done a lot of planning to make sure his teams wins. He is worried about only one opponent - Jatin, who is very good batsman.  
  
Raghav has figured out 3 types of bowling techniques, that could be most beneficial for dismissing Jatin. He has given points to each of the 3 techniques.  
  
You need to tell him which is the maximum point value, so that Raghav can select best technique.  
  
3 numbers are given in input. Output the maximum of these numbers.

#include<stdio.h>

int main() {

int n1,n2,n3,max;

scanf("%d %d %d ",&n1,&n2,&n3);

if (n1>=n2 && n1>=n3) {

max=n1;

}

else if(n2>=n1 && n2>=n3) {

max=n2;

}

else {

max=n3;

}

printf("%d\n",max);

return 0;

}

|  | **Input** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- |
|  | 8 6 1 | 8 | 8 |  |

**WEEK3.1**

1.In this challenge, we're getting started with conditional statements.  
  
Task  
  
Given an integer, *n*, perform the following conditional actions:  
  
· If *n* is odd, print Weird  
· If *n* is even and in the inclusive range of *2* to *5*, print *Not Weird*· If *n* is even and in the inclusive range of *6* to *20*, print *Weird*· If *n* is even and greater than *20*, print *Not Weird*Complete the stub code provided in your editor to print whether or not *n* is weird.

#include<stdio.h>

int main() {

int n;

scanf("%d",&n);

if(n%2!=0||(n>6&&n<=20))

printf("Weird");

else

printf("Not Weird");

return 0;

}

|  | Input | Expected | Got |  |
| --- | --- | --- | --- | --- |
|  | 3 | Weird | Weird |  |
|  | 24 | Not Weird | Not Weird |  |

Question text

2.Write a program to read two integer values and print true if both the numbers end with the same digit, otherwise print false.  
  
Example: If 698 and 768 are given, program should print true as they both end with 8.

#include<stdio.h>

int main() {

int a,b;

scanf("%d %d",&a,&b);

if(a%10==b%10) {

printf("true\n");

}

else{

printf("false\n");

}

return 0;

}

|  | Input | Expected | Got |  |
| --- | --- | --- | --- | --- |
|  | 25 53 | false | false |  |
|  | 27 77 | true | true |  |

3.Three numbers form a Pythagorean triple if the sum of squares of two numbers is equal to the square of the third.  
  
For example, 3, 5 and 4 form a Pythagorean triple, since 3\*3 + 4\*4 = 25 = 5\*5  
  
You are given three integers, a, b, and c. They need not be given in increasing order. If they form a Pythagorean triple, then print "yes", otherwise, print "no". Please note that the output message is in small letters.

#include<stdio.h>

int main() {

int a,b;

scanf("%d %d",&a,&b);

if(a%10==b%10) {

printf("true\n");

}

else{

printf("false\n");

}

return 0;

}

|  | Input | Expected | Got |  |
| --- | --- | --- | --- | --- |
|  | 3  5  4 | yes | yes |  |
|  | 5  8  2 | no | no |  |

**WEEK3.2**

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.

#include<stdio.h>

int main() {

int sides;

scanf("%d",&sides);

switch(sides) {

case 3: printf("Triangle\n");break;

case 4: printf("Square\n");break;

case 5: printf("Pentagon\n");break;

case 6: printf("Hexagon\n");break;

case 7: printf("Heptagon\n");break;

case 8: printf("Octagon\n");break;

case 9: printf("Nonagon\n");break;

case 10: printf("Decagon\n");break;

case 11: printf("The number of sides is not supported.");break;

return 0;

}

}

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

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.

#include<stdio.h>

int main() {

int year;

scanf("%d", &year);

int zodiac=(year-2000)%12;

if(zodiac<0)

zodiac+=12;

switch(zodiac) {

case 0: printf("Dragon");break;

case 1: printf("Snake");break;

case 2: printf("Horse");break;

case 3:printf("Sheep");break;

case 4: printf("Monkey");break;

case 5: printf("Rooster");break;

case 6: printf("Dog");break;

case 7: printf("Pig");break;

case 8: printf("Rat");break;

case 9: printf("Ox");break;

case 10: printf("Tiger");break;

case 11: printf("Hare");break;

default: printf("InvalidYear");

}

return 0;

}

|  | Input | Expected | Got |  |
| --- | --- | --- | --- | --- |
|  | 2004 | Monkey | Monkey |  |
|  | 2010 | Tiger | Tiger |  |

3.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.

#include<stdio.h>

int main() {

char ch;

int a;

scanf("%c%d", &ch,&a);

if((ch+a)%2==1) {

printf("The square is white.");

}

else {

printf("The square is black.");

}

return 0;

}

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

**WEEK3.3**

1.Some data sets specify dates using the year and day of year rather than the year, month, and day of month. The day of year (DOY) is the sequential day number starting with day 1 on January 1st.  
  
There are two calendars - one for normal years with 365 days, and one for leap years with 366 days. Leap years are divisible by 4. Centuries, like 1900, are not leap years unless they are divisible by 400. So, 2000 was a leap year.  
  
To find the day of year number for a standard date, scan down the Jan column to find the day of month, then scan across to the appropriate month column and read the day of year number. Reverse the process to find the standard date for a given day of year.  
Write a program to print the Day of Year of a given date, month and year.

#include<stdio.h>

int isleap(int year) {

if(year%400==0) {

return 1;

}

if(year%100==0) {

return 0;

}

if(year%4==0) {

return 1;

}

return 0;

}

int main() {

int day,month,year;

int dayofyear =0;

int dayinmonth[]={0,31,28,31,30,31,30,31,31,30,31,30,31};

scanf("%d %d %d", &day, &month, &year);

if(isleap(year)) {dayinmonth[2] = 29;}

for(int i=1; i<month;i++) {dayofyear+=dayinmonth[i];}

dayofyear+= day;

printf("%d\n", dayofyear);

return 0;

}

|  | **Input** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- |
|  | 18  6  2020 | 170 | 170 |  |

2.Suppandi is trying to take part in the local village math quiz. In the first round, he is asked about shapes and areas. Suppandi, is confused, he was never any good at math. And also, he is bad at remembering the names of shapes. Instead, you will be helping him calculate the area of shapes.  
  
· When he says rectangle he is actually referring to a square.   
· When he says square, he is actually referring to a triangle.  
· When he says triangle he is referring to a rectangle.  
· And when he is confused, he just says something random. At this point, all you can do is say 0.  
  
Help Suppandi by printing the correct answer in an integer.

#include<stdio.h>

int main() {

char shape;

int side1,side2;

int area;

scanf( "%c %d %d", &shape, &side1, &side2);

switch(shape) {

case 'C': area=side1+1%side2; break;

case 'G': area=side1-side2; break;

case 'R': area=side1\*side2; break;

case 'S': area=(side1\*side2)/2; break;

case 'T': area=side1\*side2; break;

default : area=0; break;

}

printf("%d\n",area);

return 0;

}

|  | **Input** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- |
|  | T  10  20 | 200 | 200 |  |
|  | S  30  40 | 600 | 600 |  |
|  | B  2  11 | 0 | 0 |  |
|  | R  10  30 | 300 | 300 |  |
|  | S  40  50 | 1000 | 1000 |  |

3.Superman is planning a journey to his home planet. It is very important for him to know which day he arrives there. They don't follow the 7-day week like us. Instead, they follow a 10-day week with the following days:  
  
Day Number Name of Day  
  
1 Sunday  
2 Monday  
3 Tuesday  
4 Wednesday  
5 Thursday  
6 Friday  
7 Saturday  
8 Kryptonday  
9 Coluday  
10 Daxamday  
  
Here are the rules of the calendar:  
  
• The calendar starts with Sunday always.  
• It has only 296 days. After the 296th day, it goes back to Sunday.  
  
You begin your journey on a Sunday and will reach after n. You have to tell on which day you will arrive when you reach there.

#include<stdio.h>

int main() {

int n,days,res;

scanf("%d",&n);

days=n%296;

res=days%10+1;

switch(res)

case 1: printf("Sunday");break;

case 2: printf("Monday");break;

case 3: printf("Tuesday");break;

case 4: printf("Wednesday");break;

case 5: printf("Thursday");break;

case 6: printf("Friday");break;

case 7: printf("Saturday");break;

case 8: printf("Kryptonday");break;

case 9: printf("Coluday");break;

case 10: printf("Daxamday");break;

}

return 0;

}

|  | **Input** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- |
|  | 7 | Kryptonday | Kryptonday |  |
|  | 1 | Monday | Monday |  |