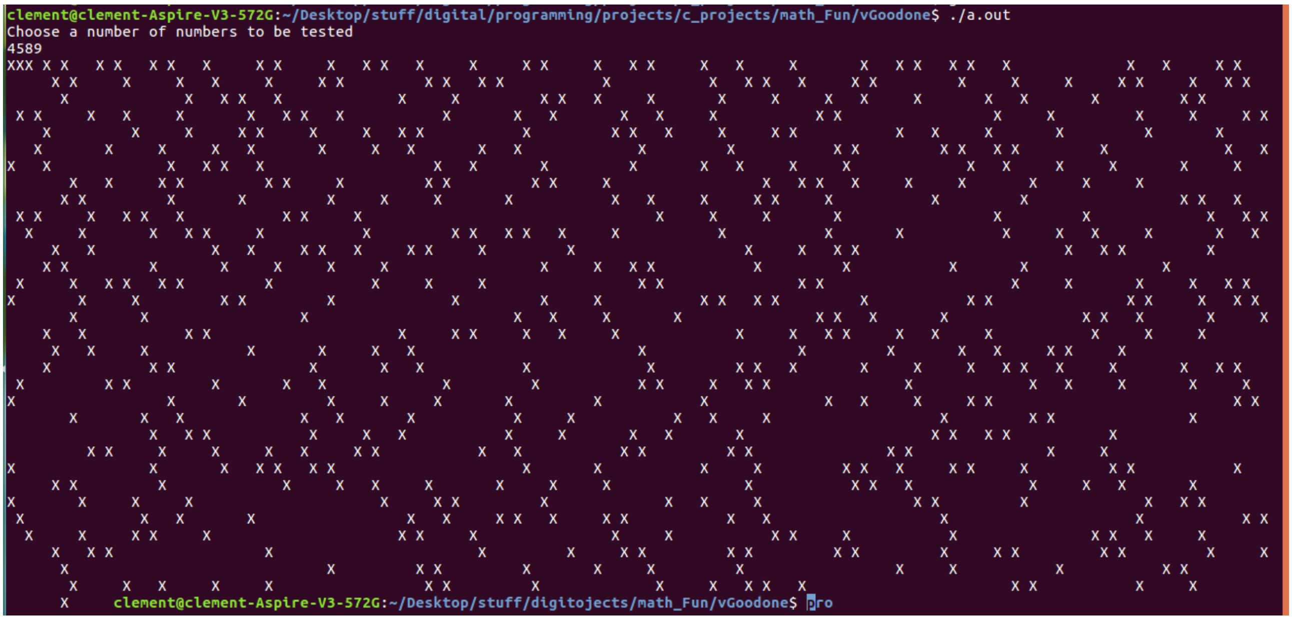
Prime numbers ‘snowFall’

Idea of the Program : to print the distribution of prime numbers out of a given set of integers
For EG : let’s take one set : [5;10] then the output will be the primes 5,7 and shall be printed out on a terminal + a text file.

General behavior for the piece of code : takes a user input for the limit of the set [1;Limit] then the program prints on a terminal ‘x’ if the number is a prime and ‘ ‘ space character if Not.

Take a screenshot as example there :



```
clement@clement-Aspire-V3-572G:~/Desktop/stuff/digital/programming/projects/c_projects/math_Fun/vGoodone$ ./a.out
Choose a number of numbers to be tested
4589
```

```
      X     clement@clement-Aspire-V3-572G:~/Desktop/stuff/digitojects/math_Fun/vGoodone$ pro
```

Is really pretty right ?

2 Steps inside this programm : Printing and calculating
-Printing: using fprintf, scanf and printf
-Calculating: Defined a C function that sort outs prime numbers within the chosen set of integers
2 sources codes for the programm above
Num one

```
#include<stdio.h>

/*This piece of program tests if a number is prime if yes prints 'X' if not prints 'space' the number of numbers to be tested is a user input defined param*/

int toCheckifPrime(int a);

int main(){

        /*Displays 'c' number of numbers to be tested*/
        int c;
        FILE *fp;
        printf("Choose a number of numbers to be tested\n");
        scanf("%d",&c);
```

```
    fp=fopen("prime_output.txt","w");

    /*Prints the output in prime_output.txt*/
    for(int l=1;l<c;l++){
        fprintf(fp,"%c",toCheckifPrime(l));
    /*Skips a line every 100 characters*/
        if(l%100==0){
            fprintf(fp,"\n");
        }
    }
    /*Prints the output in the user terminal*/
    for(int l=1;l<c;l++){
        printf("%c",toCheckifPrime(l));
    /*Skips a line every 100 characters*/
        if(l%100==0){
            fprintf(fp,"\n");
        }
    }

 return 0;

}
//Ascii code space : 32, 'X' : 88
int toCheckifPrime(int a){

    int inter=0; short b=0;
    for(int i=2;i<a;i++){
        inter=a%i;
    //printf("%d\n",inter);
        if(inter==0){
            //printf("TesteHere\n");
            return 32;
            b=1;
            break;
        }

    }
    if(b!=1)return 88;


}
```

Num 2

```
#include<stdio.h>
#include<stdbool.h>

/*This piece of program tests if a number is prime using modulo poerator (%) if yes prints 'x' if not
prints 'space' the number of numbers to be tested is a user input defined param*/

bool toCheckifPrime(int a);
```

```
int main(){

    /*Displays 'c' number of numbers to be tested*/
    int c;
    FILE *fp;
    printf("Choose a number of numbers to be tested\n");
    scanf("%d",&c);
    fp=fopen("prime_output.txt","w");

    /*Prints the output in prime_output.txt*/
    for(int l=1;l<c;l++){
    if(toCheckifPrime(l)==true){
            fprintf(fp,"%s","x");

        }
        else{
            fprintf(fp,"%s"," ");
        }

    /*Skips a line every 100 characters*/
        if(l%100==0){
            fprintf(fp,"%s","\n");
        }

    }
    /*Prints the output in the user terminal*/
    for(int i=1;i<c;i++){
        if(toCheckifPrime(i)==true){
            printf("x");

        }
        else{
            printf(" ");
        }

    /*Skips a line every 100 characters*/
        if(i%100==0){
            printf("\n");
        }
    }

 return 0;

}



/*Function returns True is the integer input is a prime number, false otherwise integer >=0*/
bool toCheckifPrime(int a){
    bool outPut=true;
    int inter=0;
    for(int i=2;i<a;i++){
```

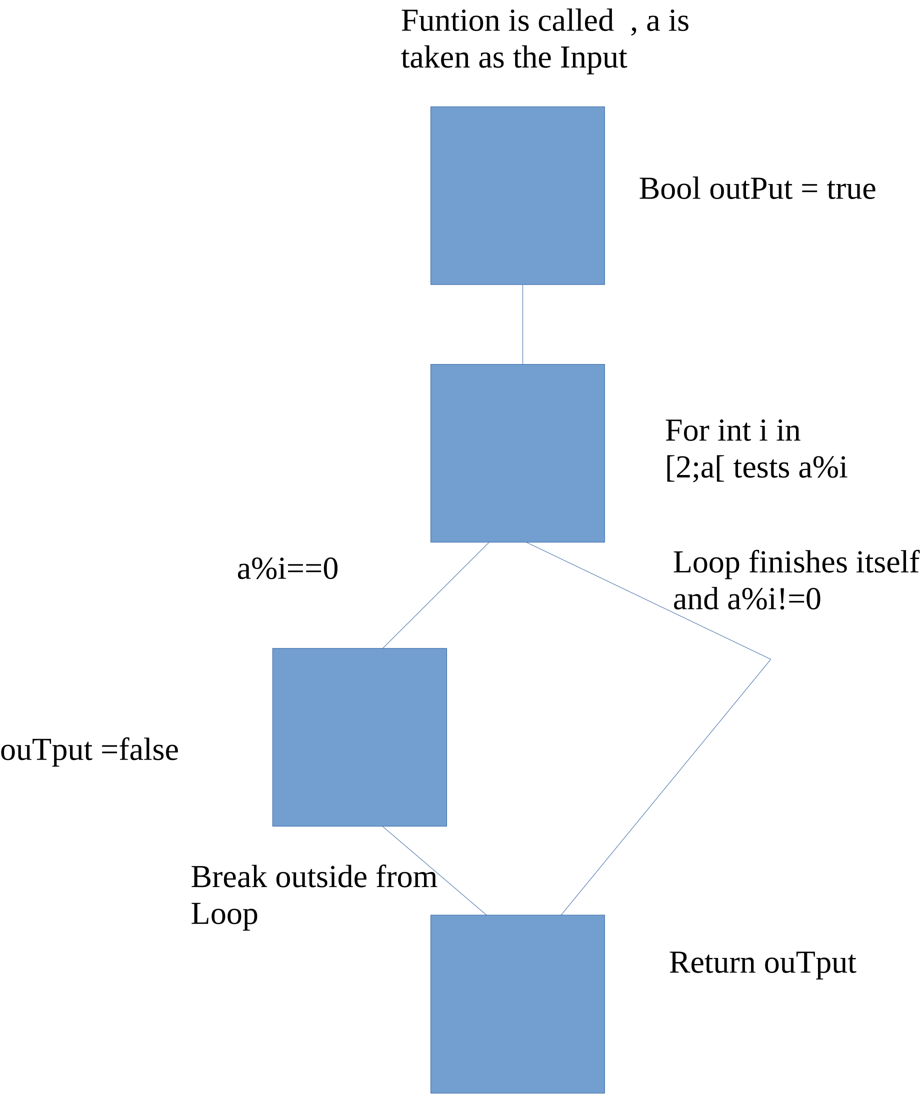
```
                inter=a%i;
                if(inter==0){
                        outPut=false;
                        break;
                }

        }
        return outPut;
}
```

Function that calculates the prime
idea of Function : tests in a for loop all divisions (modulo % operator) between 2 and the number itself : [2;N[where N is the number chosen. -*Because a number is a Prime if and only if its divisors are 1 and itself; no more.*-

Schematics of the function :

Int a
Bool ouTput


Funtion is called , a is taken as the Input
Bool outPut = true
For int i in [2;a[tests a%i
a%i==0
Loop finishes itself and a%i!=0
ouTput =false
Break outside from Loop
Return ouTput

Difference between the two versions

One returns true only if the number is prime this is logically solid and handy that’s Num One
The other returns ASCII codes one for A prime and another one for not a prime, this version is less rigorous logically but make the code more compact because it modify and simplify the display part

Function and libraries >Used :
stdio : scanf, fprintf, printf
stbool:
Variables defined for this program :
int inter
loop varaibles
int c : user
FILE *fp
bool outPut

File output and lines skipping params

The progamm prints out a text file’prime_snowFall.txt’ in the current folder and prints everything inside as for display options skipped lines every 10 lines This is hard coded at this inside the programm :

```
/*Skips a line every 100 characters*/
                    if(l%100==0){
                              fprintf(fp,"\n");
                    }
```

Note : modulo “%” operator Behavior
The result doesn’t make any sense if a<b “a%b”, One need to be careful about it Sould be a remainder of a division between integers.
5%3=2
3%5=3 Doesn’t mean anything here for our program
10%5=0

Source : https://www.geeksforgeeks.org/modulo-operator-in-c-cpp-with-examples/