¿Qué estudiamos en este curso?

- Vamos a estudiar problemas físicos realizables computacionalmente.
- Nos enfocamos en observables estadísticos (ensambles).
- Analizamos las limitaciones de simular sistemas " infinitos" por medio de réplicas "finitas"
- NO es un curso de "cálculo numérico".

Tres problemas a atacar:

1. Percolación

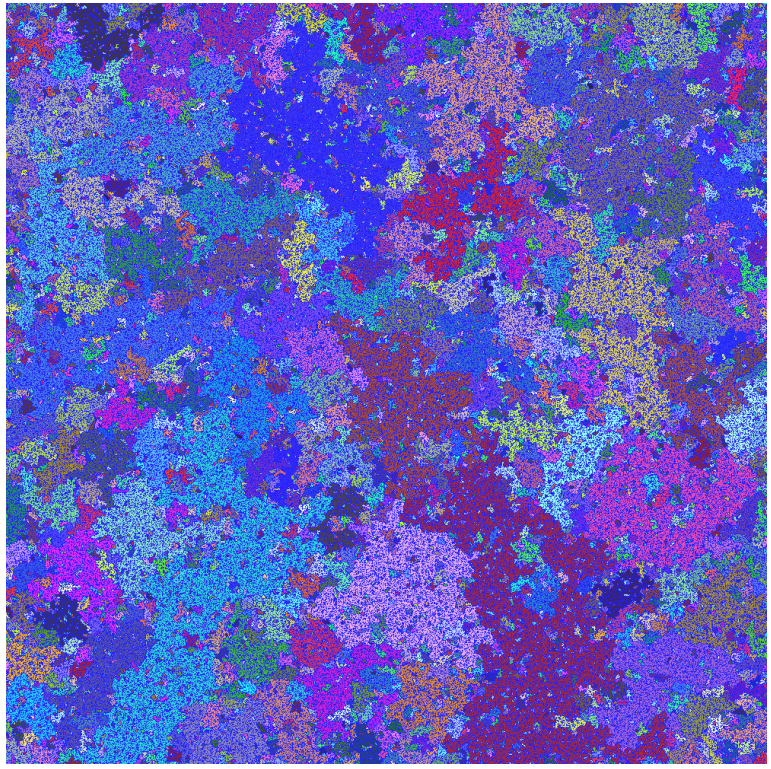
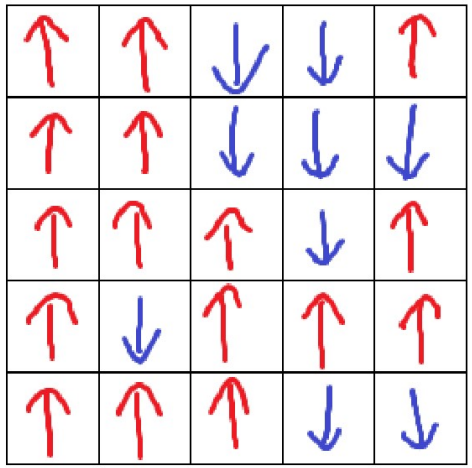


Figure: Sistema percolante.

- Es el más "simple" de todos porque consiste en partículas "quietas y sin interacciones entre sí".

Tres problemas a atacar:

2. Problema de Ising 2D



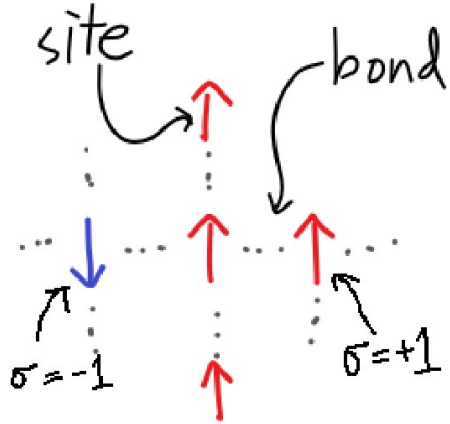
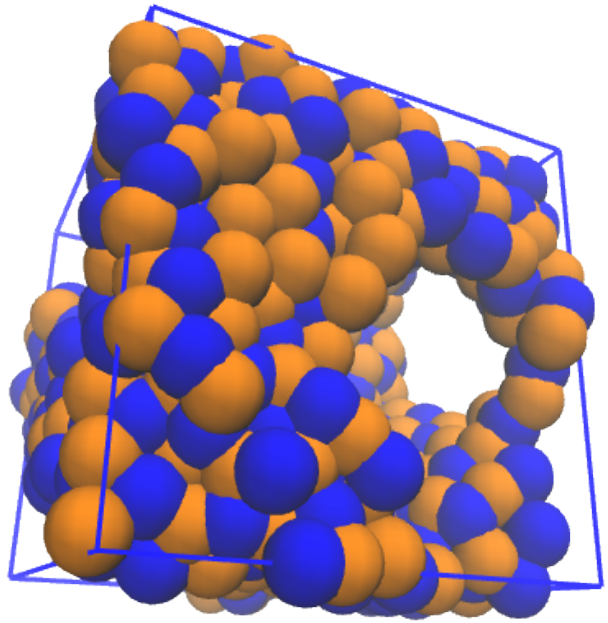


Figure: Modelo de Ising (2D).

- Es "moderadamente simple" porque consiste en partículas "quietas con interacciones entre sí".

Tres problemas a atacar:

3. Dinámica molecular



- Es el más “complejo” de todos porque consiste en partículas “con momento y con interacciones entre sí”.

¿Cómo atacamos estos problemas?

- Los sistemas de "muchas partículas" requieren mucho esfuerzo computacional.
- $\Rightarrow$ Lenguajes Fortran, C, C++, etc.
- NO es factible usar lenguajes "interpretados": MatLab, Python, Matemática, R, etc.... salvo para el post-procesamiento.

Intro al lenguaje C: estructura

```
// primero comenzamos incluyendo las librerías

#include <stdio.h>
#include <stdlib.h>
#include <math.h>

// luego sigue el programa principal

int main()
{

...

  return 1;
}
```

Intro al lenguaje C: variables

```
// en cada rutina se definen variables

int main()
{

  int i;
  float a;
  double p,red[4096]; // red es un vector.


....


  return 1;
}
```

Intro al lenguaje C: loops

```
int main()
{

  int i;
  float a;
  double p,red[4096]; // red es un vector.

  p = 0.0;


  for (i=0;i<4096;i++)// loop en C. i++ <--> i=i+1
    {
      red[i] = 0.0;
    }


  return 1;
}
```

Intro al lenguaje C: loops

```
int main()
{

  int i;
  float a;
  double p,red[4096]; // red es un vector.

  i = 0;


  while (i<4096)
    {
      red[i] = 0.0;
      i++;
    }


  return 1;
}
```

Intro al lenguaje C: constantes

```
#define N 4096
#define SEED 260572

int main()
{

  int i;
  float a;
  double p,red[N]; // red es un vector.

  p = 0.0;


  for (i=0;i<N;i++) red[i] = 0.0;


  return 1;
}
```

Intro al lenguaje C: números pseudo-aleatorios

```
#define SEED 260572

int main()
{

  int i;
  double p;

  srand(SEED);

  p = (double)rand()/(double)RAND_MAX;


  printf(``%lf\n'',p);


  return 1;
}
```

Intro al lenguaje C: números con probabilidad dada

```
#define N 100
#define P 0.7
#define SEED 260572

int main()
{
  int i;
  double p,ratio=0.0;

  srand(SEED);

  for(i=0;i<N;i++)
    {
      p = (double)rand()/(double)RAND_MAX;
      if (p<P) ratio += 1.0/(double)N;
    }
  return 1;
}
```

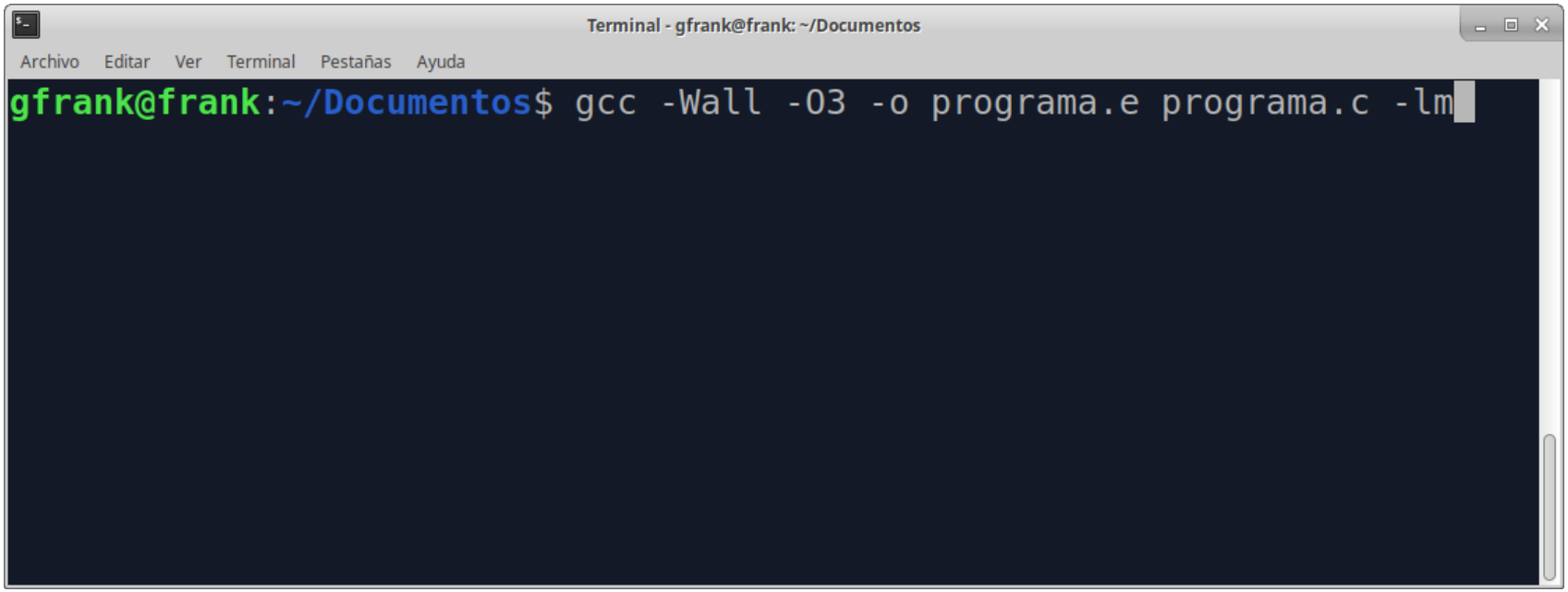
Intro al lenguaje C: funciones en C

```c
#include <stdio.h>
#include <stdlib.h>
#include <math.h>

int llenar(double *red,double p);

int main()
{
  llenar(red,p);
  return 1;
}

void llenar(double *red,double p)
{
  ...
}
```

Intro al lenguaje C: función `llenado()`

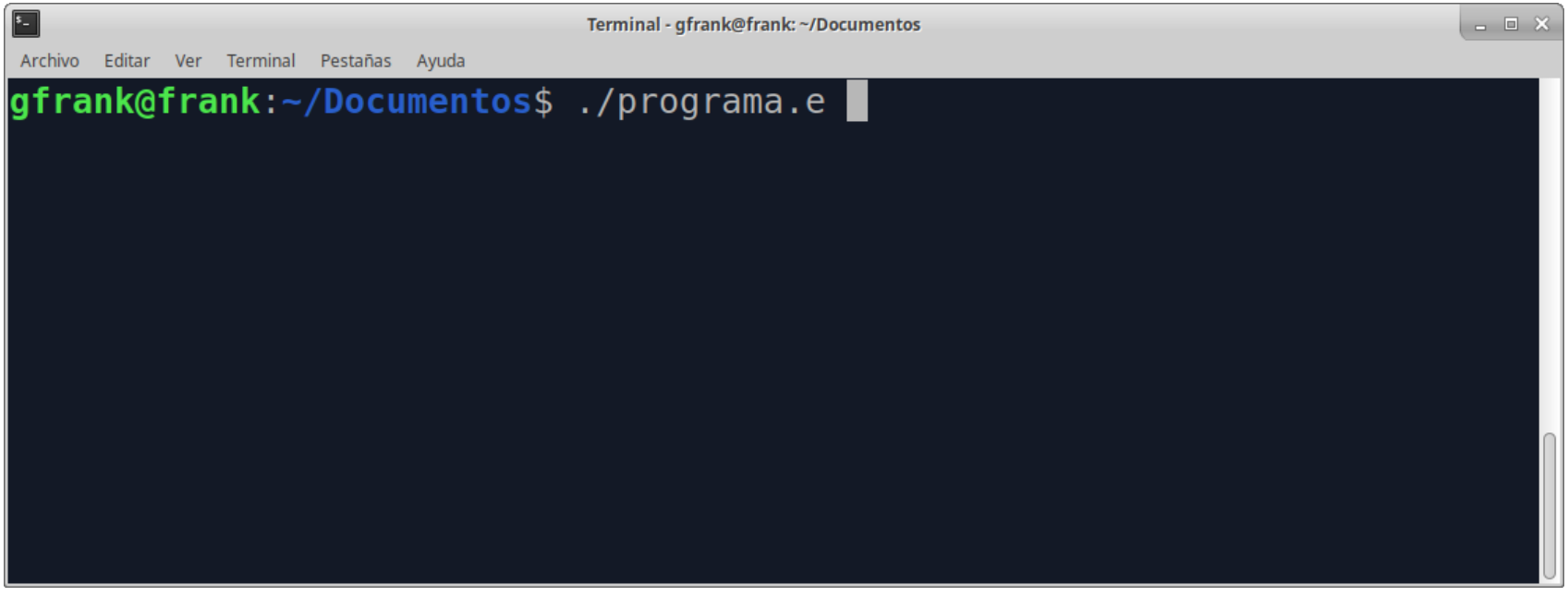
```
void llenar(double *red,double p)
{
  int i;
  double ratio;

  for(i=0;i<N;i++)
    {
      ratio = (double)rand()/(double)RAND_MAX;

      if (ratio<p) red[i] = 1;
      else red[i] = 0;
    }
  return;
}
```

Intro al lenguaje C: compilación



- Usamos el compilador gcc, pero bien podemos usar otro!

Intro al lenguaje C: compilación



- Corrremos....