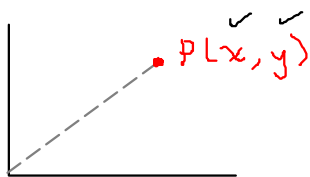


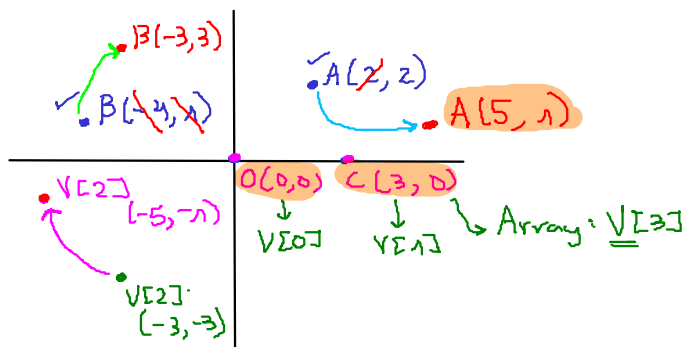
23/09/2021.

Estructuras - Repaso



```
struct _Punto {  
    double x,  
    double y  
};
```

```
typedef struct _Punto Punto;
```



```
Punto A = { 2, 2 };  
Punto B = { .x = -4,  
            .y = 1 };
```

```
Punto O, C;  
O.x = 0;  
O.y = 0;  
C.x = 3;  
C.y = 0;
```

```
Punto V[3];  
V[0] = O;  
V[1] = C;  
V[2].x = -3;  
V[2].y = -3;
```

```
Punto *ptr1 = &A;
```

```
Punto *ptr2;  
ptr2 = &V[2];
```

```
B.x = -3;  
B.y = -3;
```

} variable.miembro

```
ptr1->x = 5; // A.x = 5  
ptr1->y = 1; // A.y = 1
```

{ apuntador -> miembro

```
V[2].x = -5;  
ptr2->y = -1;
```

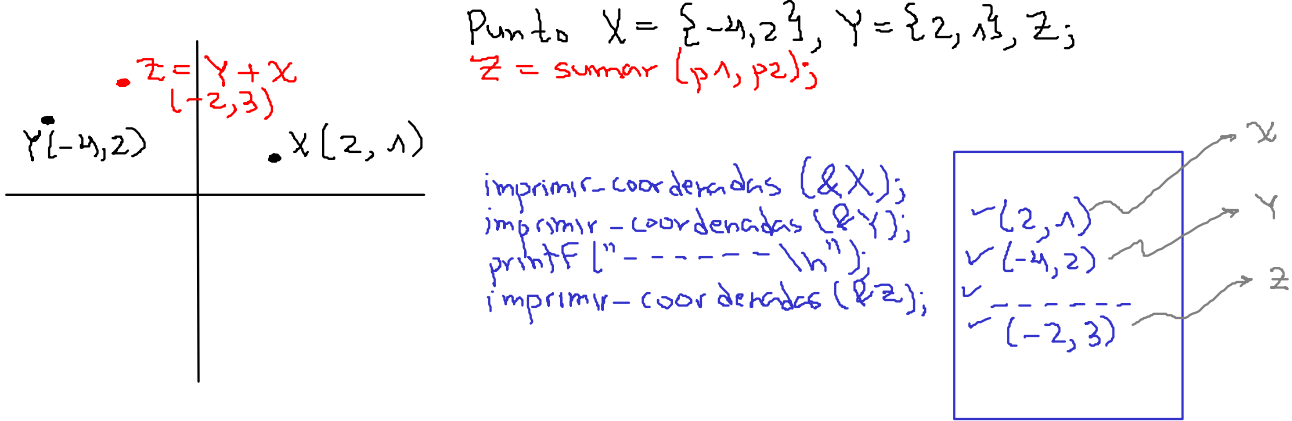
```
ptr2 = ptr1; // ptr2 = &A;
```

Funciones:

```
void imprimir_coordenadas (Punto *p) {  
    printf ("%lf , %lf\n", p->x, p->y);  
}
```

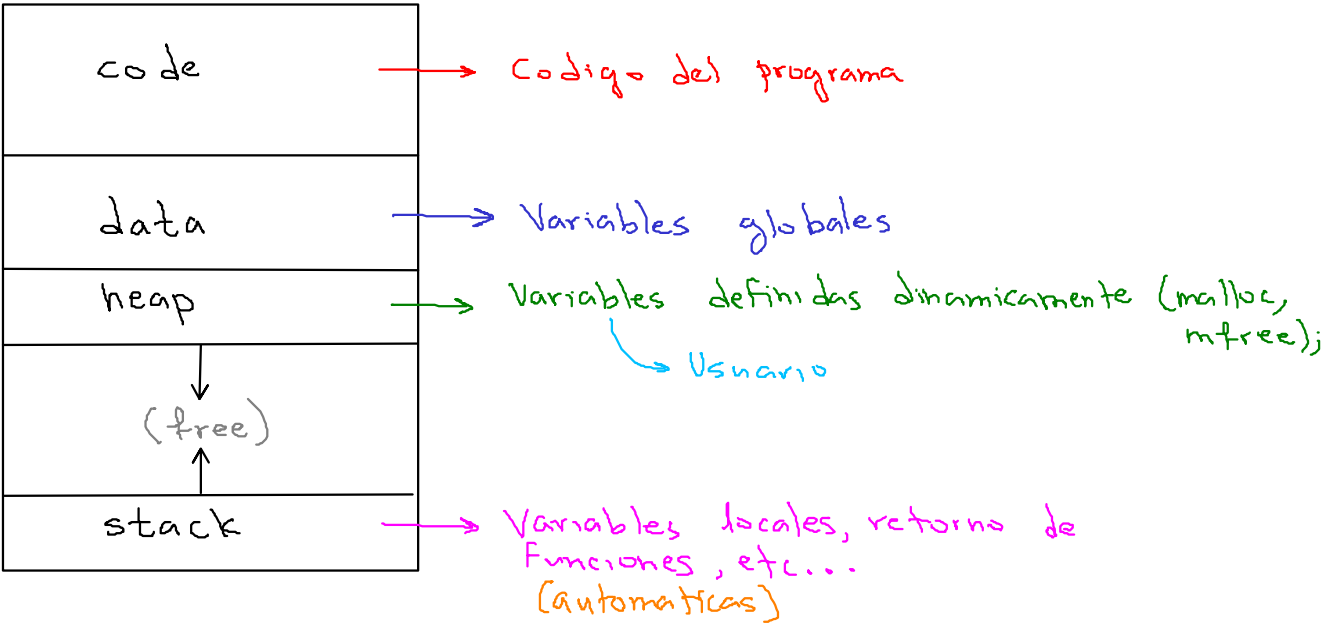
```
Punto sumar (Punto p1, Punto p2) {
    Punto s;
    s = p1 + p2;
    return s;
}
```

Ejemplo usando las funciones:



Reserva dinamica de memoria.

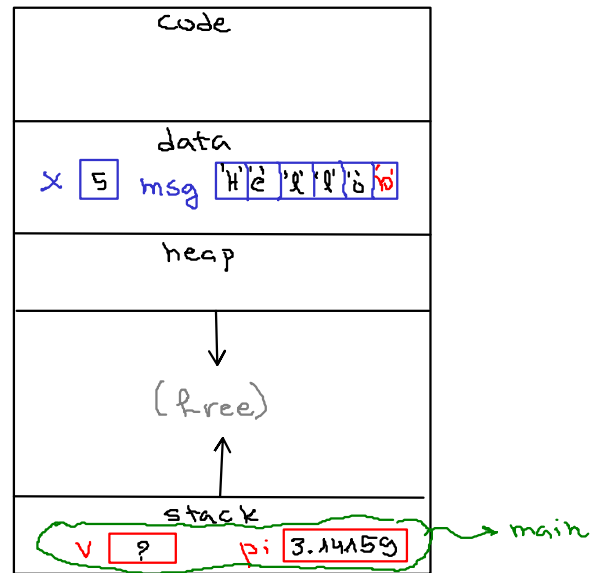
Mapa de memoria



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

```
int x=5; //global
char msg[] = "Hello"; //global
```

```
int main(int argc, const char* argv[]) {
    int v; //local
    float pi = 3.14159; //local
    printf("%d\n", x);
    printf("%f\n", pi);
    return 0;
}
```

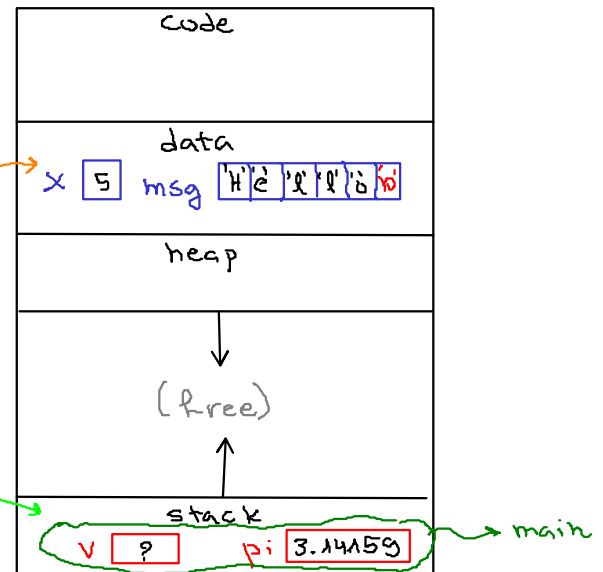
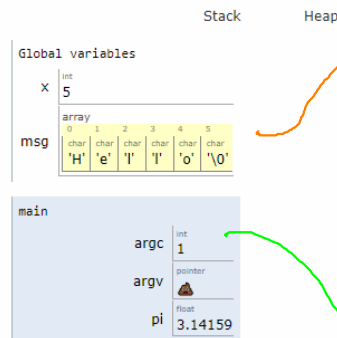


C (gcc 4.8, C11) [unsupported]

```
1 #include <stdio.h>
2
3 int x=5;
4 char msg[] = "Hello";
5
6 int main(int argc, const char* argv[]) {
7     int v;
8     float pi = 3.14159;
9     printf("%d\n", x);
10    printf("%f\n", pi);
11    return 0;
12 }
```

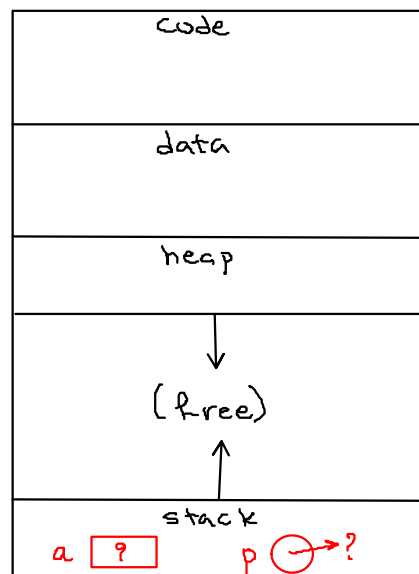
Print output (drag lower right corner to resize)

```
5
3.141590
```



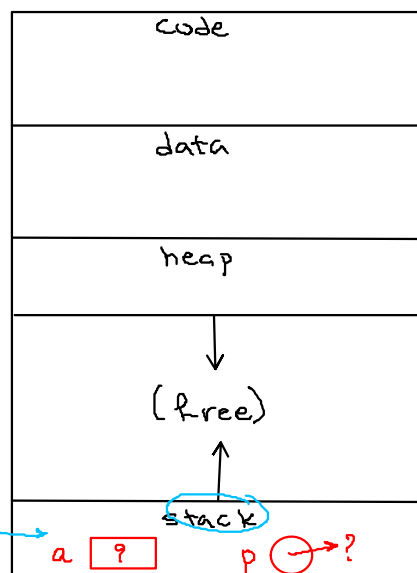
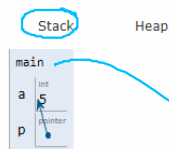
```
%%tutor -l c -k
#include <stdio.h>
```

```
int main() {
    ✓ int a; //local
    ✓ int *p = &a; //local
    *p = 5;
    printf("*p = %d\n", *p);
    return 0;
}
```



```
1 #include <stdio.h>
2
3 int main() {
4     int a;
5     int *p = &a;
6     *p = 5;
7     printf("**p = %d\n", *p);
8     return 0;
9 }
```

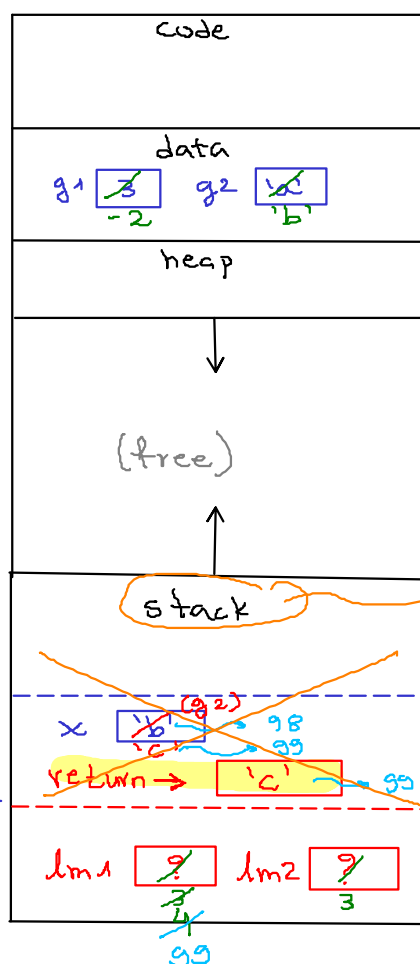
*p = 5



```

1  int g1 = 3; // global
2  char g2 = 'a'; // global
3
4  int main() {
5      int lm1, lm2; // local
6      lm1 = 3;
7      lm2 = lm1++;
8      g2++;
9      g1 -= 2;
10     lm1 = f(g2);
11
12     return 0;
13 }
14
15 int f1(int x) { // local
16     return x++;
17 }
18
19
20 int f2() {
21     g1 *= 2;
22 }

```



↑ sube
↓ baja

'b' \rightarrow 98
'c' \rightarrow 99

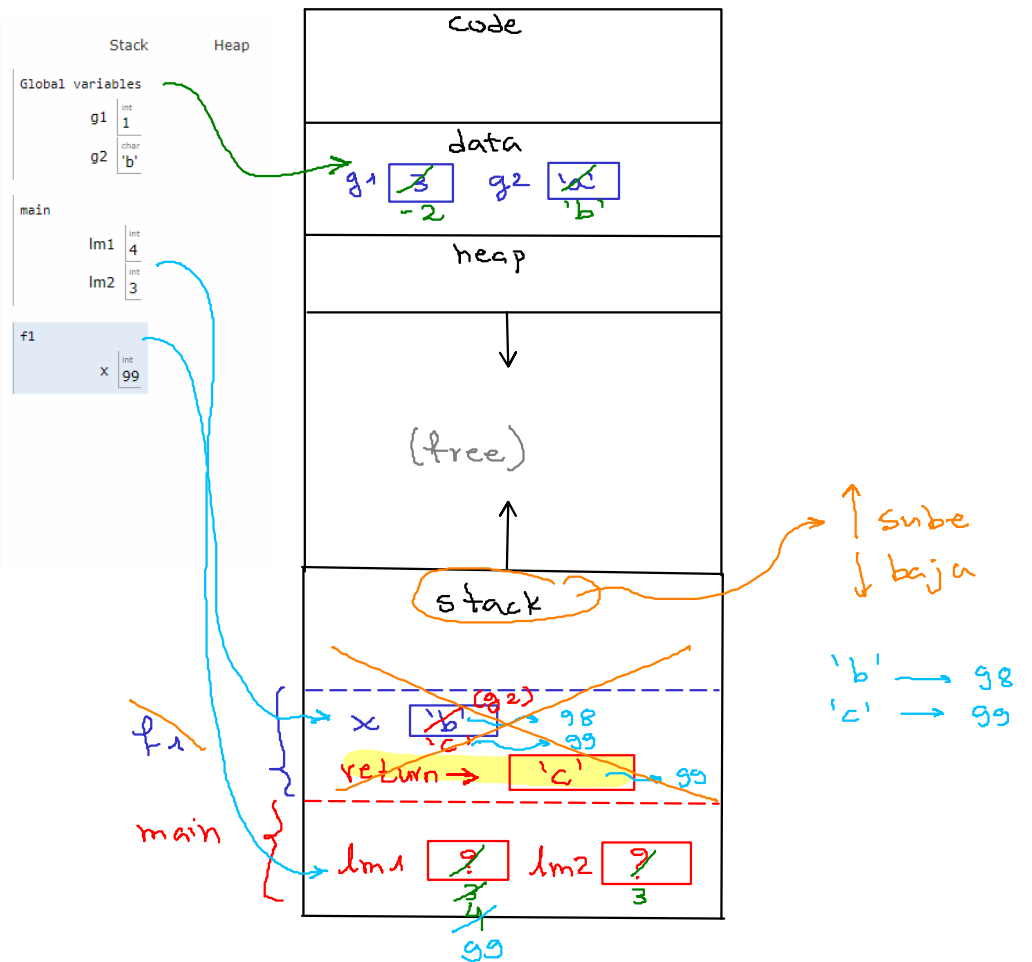
C (gcc 9.3, C17 + GNU extensions)
(known limitations)

```

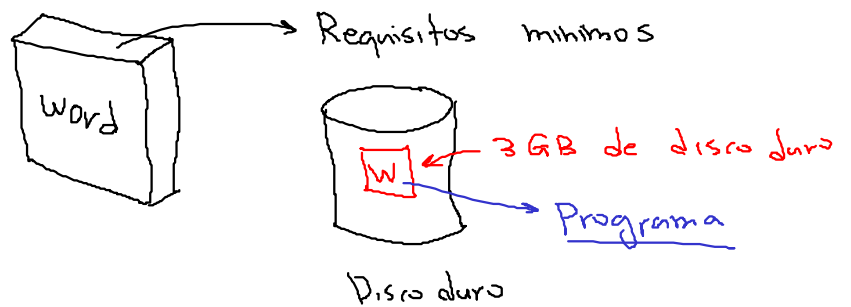
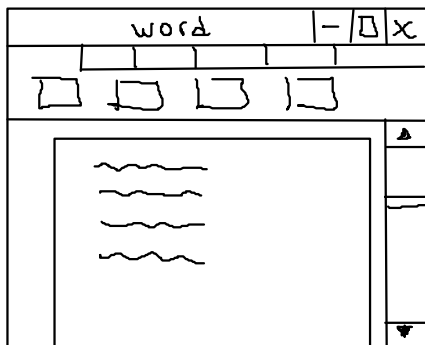
1 int g1 = 3;
2 char g2 = 'a';
3
4 int main() {
5     int lm1, lm2;
6     lm1 = 3;
7     lm2 = lm1++;
8     g2++;
9     g1 -= 2;
10    lm1 = f1(g2);
11
12    return 0;
13 }
14
15 int f1(int x) {
16     return x++;
17 }
18
19
20 int f2() {
21     int *p = 0;

```

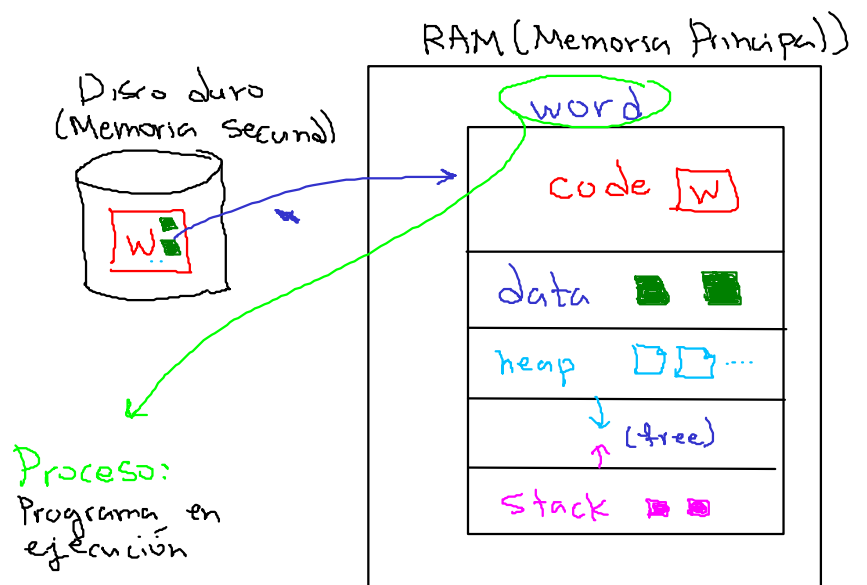
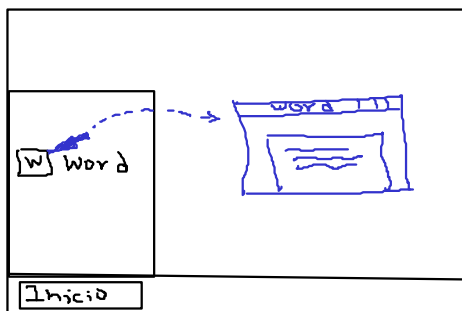
[Edit this code](#)

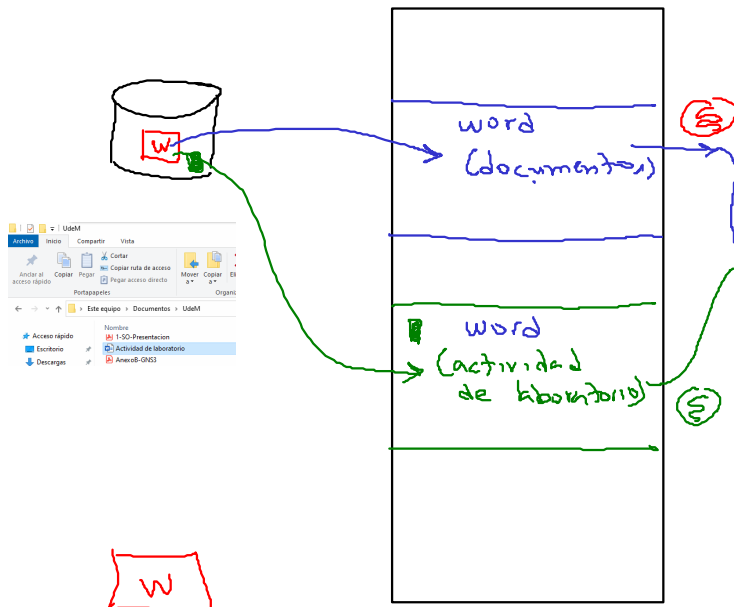


Por que es importante reservar dinamicamente memoria?



Programa ya instalado





Administrador de tareas

Archivo Opciones Vista

Procesos Rendimiento Historial de aplicaciones Inicio Usuarios Detalles Servicios

Nombre	Estado	14% CPU	41% Memoria	0% Disco	0% Red
Aplicaciones (8)					
Administrador de tareas		1.2%	25.3 MB	0 MB/s	0 Mbps
Explorador de Windows		0.1%	59.3 MB	0 MB/s	0 Mbps
Google Chrome (17)		0.7%	709.8 MB	0.1 MB/s	0.1 Mbps
Microsoft Word (2)		0%	149.4 MB	0 MB/s	0 Mbps
Actividad de laboratorio - Word					
Documento1 - Word					
Paint		0%	19.8 MB	0 MB/s	0 Mbps
xournalpp_bin		0%	77.1 MB	0 MB/s	0 Mbps
Zoom Meetings (32 bits) (2)		5.4%	153.2 MB	0 MB/s	0.5 Mbps
Zoom Meetings (32 bits)		0%	46.5 MB	0 MB/s	0 Mbps
Procesos en segundo plano (77)					
Adobe Acrobat Update Service (...)		0%	0.7 MB	0 MB/s	0 Mbps
Aislamiento de gráficos de disp...		1.6%	36.6 MB	0 MB/s	0 Mbps

Menos detalles Finalizar tarea

Hemos cargado "2 procesos".