

```
#define t_elen - matrix int
#define t_elen - queue char*
```

(2) Jpe Cruz

```
typedef struct {
    int min;
    queue* cds;
} solucion camino;
```

```
int obtener_id (char* nombre) {
    int resultado = -1;
    if (nombre == "Madrid")
        resultado = 0;
    if (nombre == "Londres")
        resultado = 1;
    if (nombre == "Roma")
        resultado = 2;
    if (nombre == "Paris")
        resultado = 3;
    return resultado;
}
```

```
char* obtener_nombre (int id) {
    char* resultado;
    if (id == 0)
        return char[10] = "Madrid";
    if (id == 1)
        return char[10] = "Londres";
    if (id == 2)
        return char[10] = "Roma";
    if (id == 3)
        return char[10] = "Paris";
    return char[10] = "Undefined";
}
```

```
camino* get_path (graph* g, char* origen, char* destino, matrix* distancias,
matrix* procedimientos) {
    if (g == NULL || distancias == NULL || procedimientos == NULL) return NULL;
    int index = -1;
    camino* way = camino_crear();
    int index_origen = obtener_id (origen);
    int index_destino = obtener_id (destino);
    queue* edesminos = queue_new();
    if (index == index_destino) {
        int aux = 0;
        enqueue (edesminos, obtener_nombre (origen));
        while (index != index_destino) {
            int index = matrix_get (procedimientos, index_origen, index_destino);
            way_enqueue (edesminos, obtener_nombre (index));
            aux = camino_get_min (way); index_origen = index;
            camino_set_min (way, aux + matrix_get (distancias,
        }
    }
    camino_set_min (way, matrix_get (distancias, index_origen, index_destino));
    camino_set_cds (way, edesminos);
    return way;
}
```

Es muy particular. Hay que usar graph_vertex_index con la función de strcmp

```
int camino_get_min (camino* un_camino) {
    return un_camino.min;
}
void camino_set_min (camino* un_camino, int minimo) {
    un_camino.min = minimo;
}
camino* crear_camino_crear() {
    queue* edesminos = queue_new();
    camino* resultado = malloc (sizeof (camino));
    resultado->cds = edesminos;
    resultado->min = 0; OJO!!!!!!
    return resultado;
}
```

```
char* camino_get_cds (camino* un_camino) {
    return un_camino->cds;
}
void camino_set_cds (camino* un_camino, queue* cds) {
    un_camino->cds = cds;
    return;
}
```