Nombre y Apellido:

Examen Parcial 1

Considere las siguientes definiciones de Lista Simplemente Enlazada General:

```
typedef void (*FuncionDestructora)(void *dato);
typedef void *(*FuncionCopia)(void *dato);
typedef void (*FuncionVisitante)(void *dato);
typedef int (*FuncionComparar)(void *dato1, void *dato2);
typedef int (*Predicado) (void *dato);

typedef struct _GNode {
  void *data;
  struct _GNode *next;
} GNode;

typedef struct _GNode *GList;

/** Devuelve una lista vacia.
*/
GList glist_crear();

/** Destruccion de la lista.
*/
void glist_destruir(GList list, FuncionDestructora);

/** Determina si la lista es vacia.
*/
int glist_vacia(GList list);

/** Agrega un elemento al inicio de la lista.
*/
GList glist_agregar_inicio(GList list, void *data, FuncionCopia);

/** Busca un elemento en la lista.
*/
int glist_buscar(GList list, void* data, FuncionComparar);

/** Recorrido de la lista, utilizando la funcion pasada.
*/
void glist_recorrer(GList list, FuncionVisitante);

/** Dada una lista, retorna una nueva lista con los elementos que cumplen con
    el predicado.
*/
GList glist_filtrar(GList, Predicado, FuncionCopia);
```

Y las siguientes definiciones extra:

```
typedef void** GArray;

typedef struct {
    char *nombre, *direccion, *dni;
    int edad;
} Persona;
```

1. Implemente una función

```
GArray listToArray (GList lista, FuncionCopia c);
```

que vuelque los datos de la lista recibida en un arreglo de tipo `GArray`. El arreglo debe ser redimensionable, adaptándose a la cantidad de elementos que posee la lista.

2. Utilice la función implementada en el ejercicio anterior para definir una nueva función

```
GArray listToArrayPersona (GList lista);
```

que tome una lista de estructuras `Persona` y devuelva un arreglo con los elementos de la lista reutilizando las estructuras (sin realizar copias físicas de las mismas).

3. Implemente una función

```
GList mayores18ConA (GList lista);
```

que, dada una lista de elementos de tipo `Persona`, utilice la función `glist_filtrar` y devuelva una nueva lista con aquellos elementos cuya edad sea mayor a 18 años y tales que en su dirección hay al menos una letra *"a"* (ya sea mayúscula o minúscula).