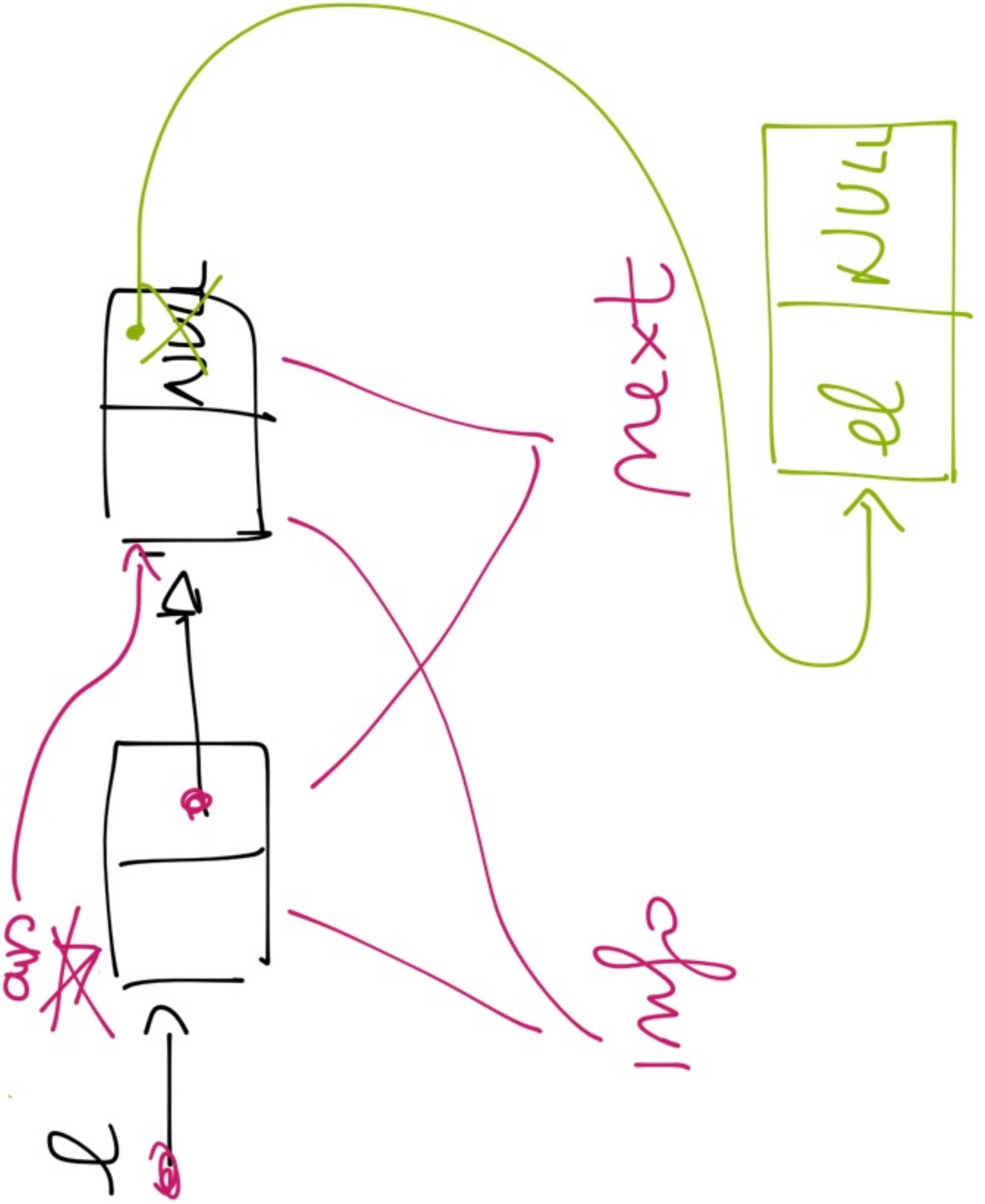
```
void aggingi (ListeDiElementi l,
              int el)
{ ListeDiElementi aus = l;
  while (aus->next != NULL)
      aus = aus -> next;
  aus->next = malloc(sizeof(ElementoLista));
  aus->next->next = NULL;
  aus->next->info = el;
}
```

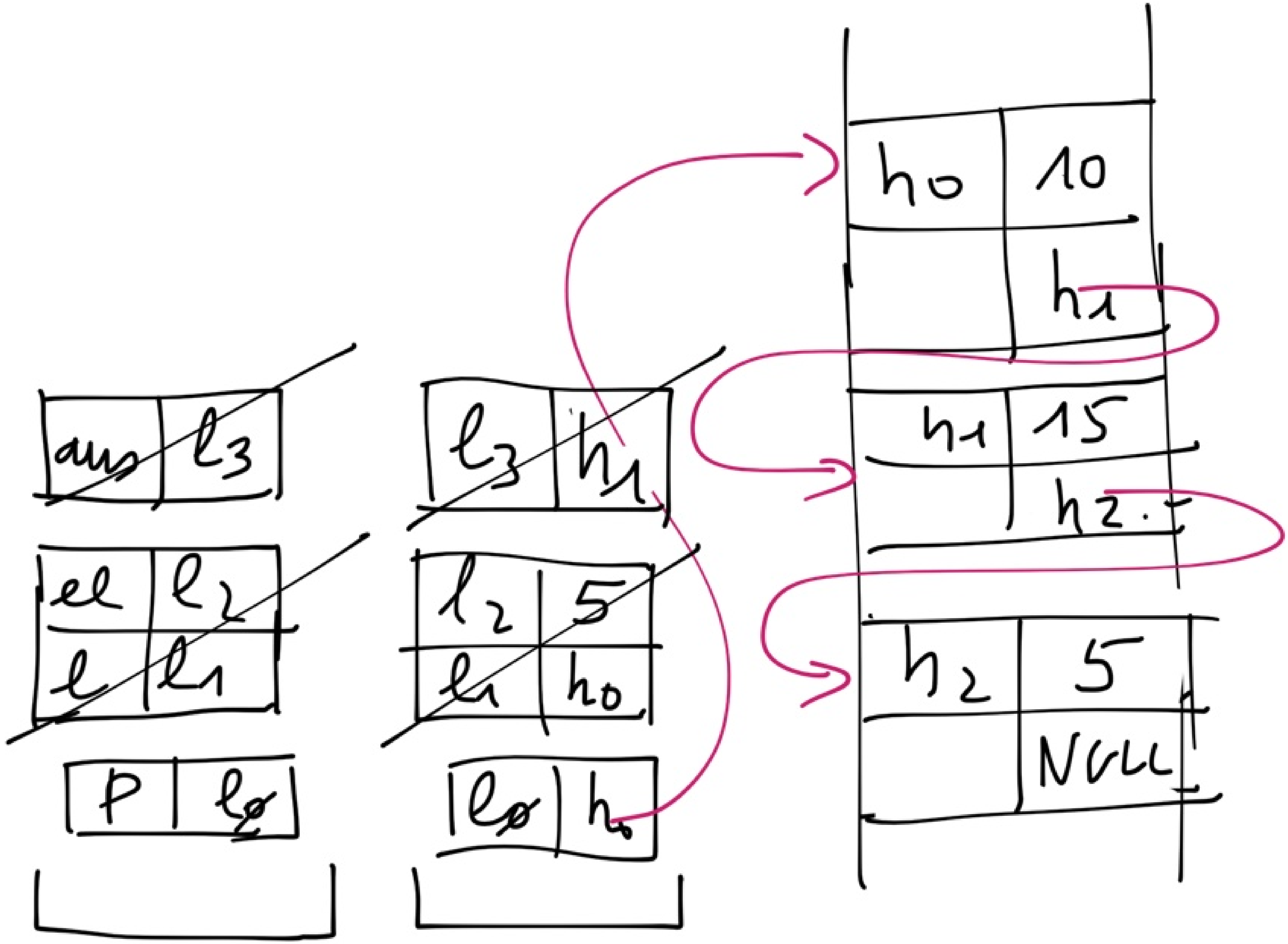
struct el { int num;
struct el *next }
Elementlista;

typedef struct el *
typedef Elementlista *
Lista;

Lista listaElementi;

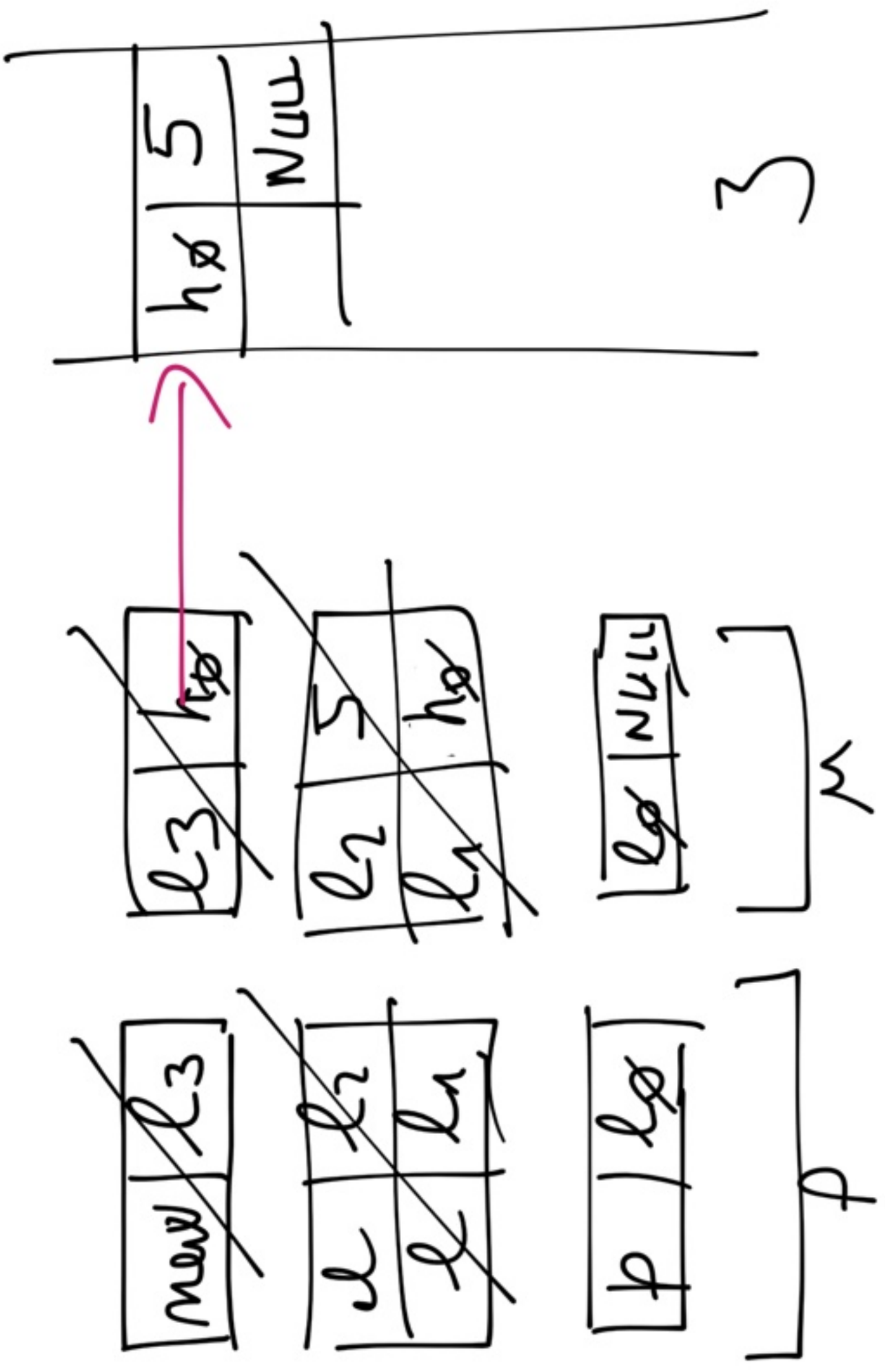


aggiungi (p,5);



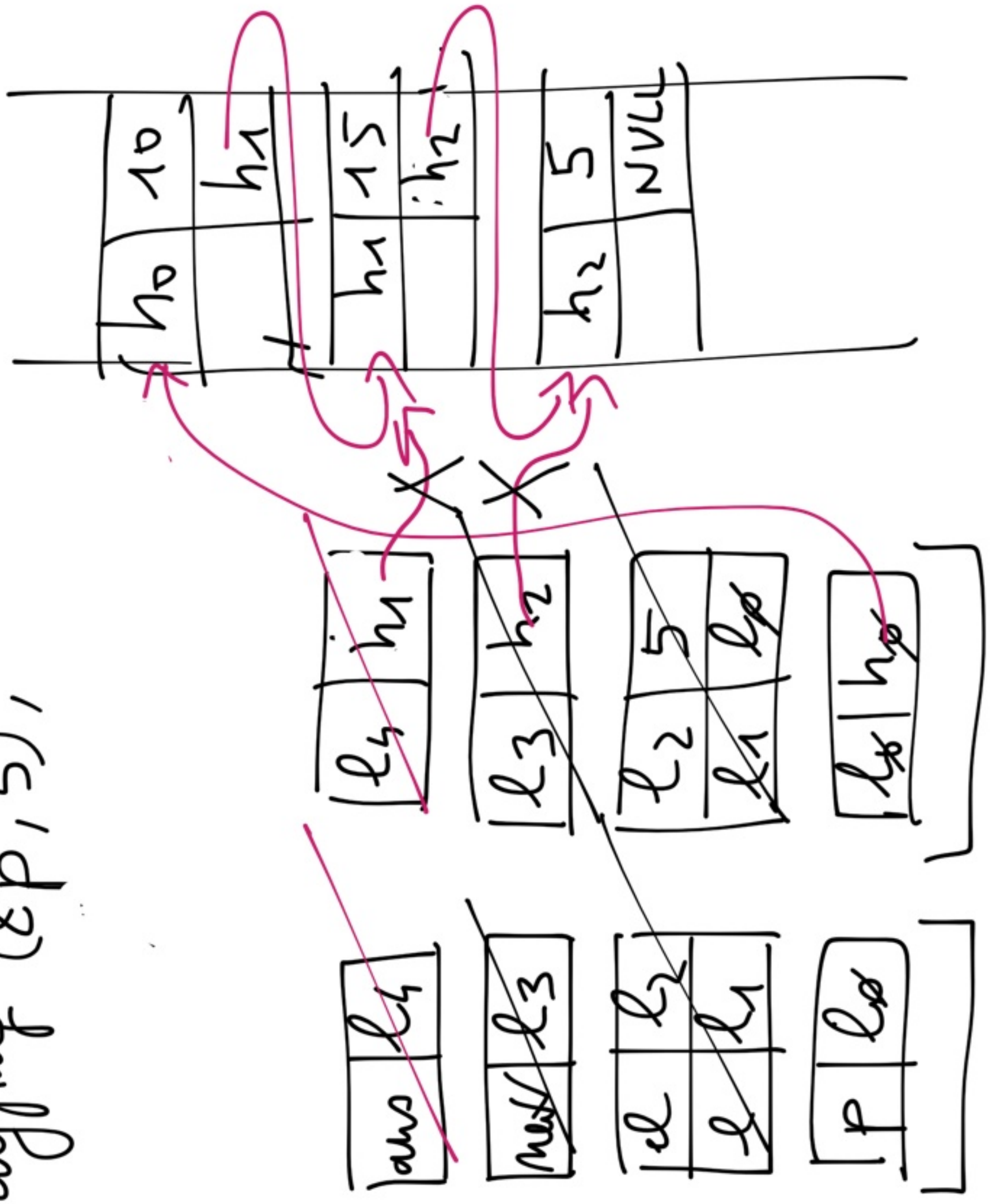
```
void aggiungi (ListaDiElementi l, int el)
{ listaDiElementi new = malloc( .... );
  new -> inf = el; new->next = NULL;
  if (l == NULL) l = new
  else
  { listaDiElementi aus = l;
    while (aus -> next != NULL)
      aus = aus -> next;
    aus -> next = new;
  }
}
```

aggiungi (p, 5);

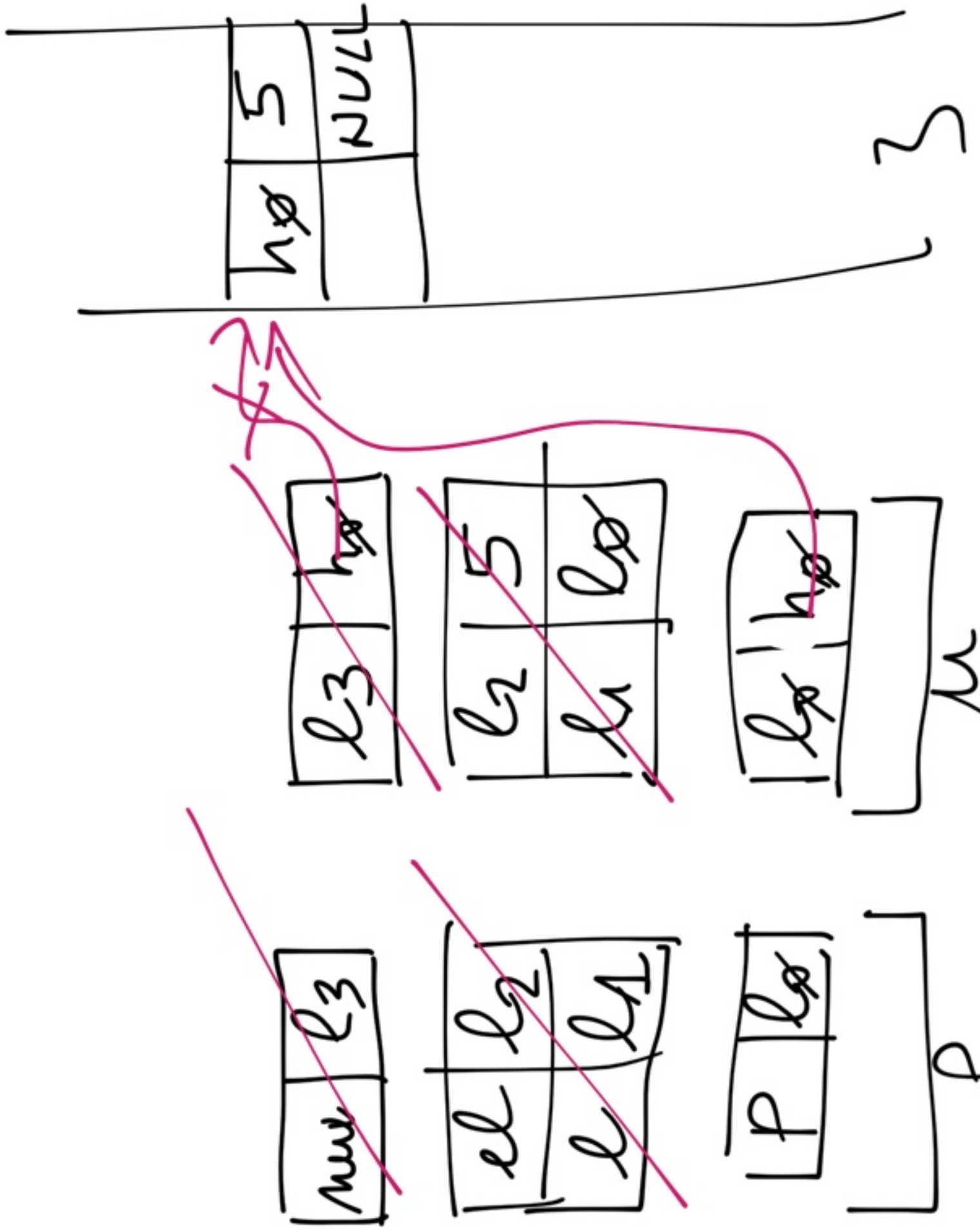


```
void aggiungi (ListaDiElementi *l, int el)
{ listaDiElementi new = malloc( .... );
  new -> inf = el; new->next = NULL;
  if (*l == NULL) *l = new;
  else
  { listaDiElementi aus = *l;
    while (aus -> next != NULL)
      aus = aus -> next;
    aus -> next = new;
  }
}
```

aggiungi(&p1, 5);



aggiungi(&p, 5);:



· Regoline semplicissime

dovete modificare il puntatore al primo elemento di una lista

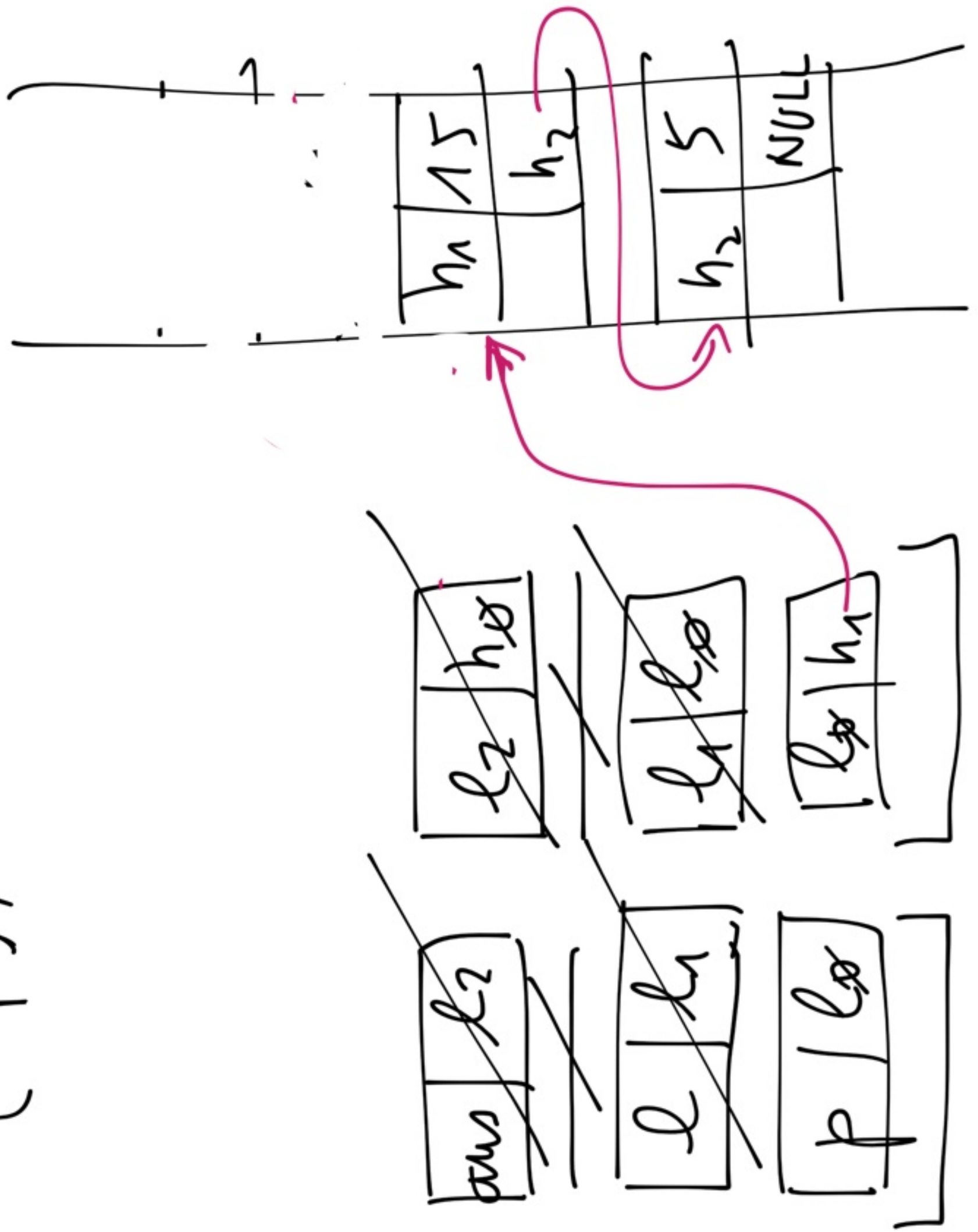
```
void p(ListaDiElementi * l,...
```

non dovete modificare il puntatore al primo elem. di una lista

```
void p(ListaDiElementi l,...
```

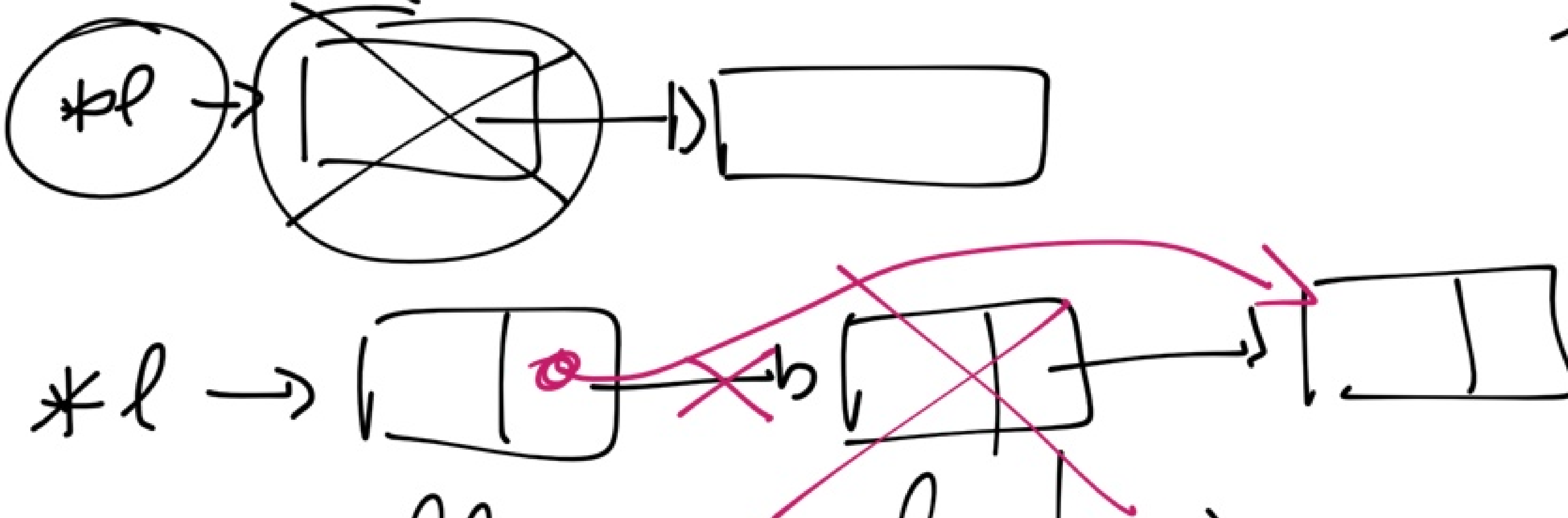
Cancellare il primo elemento di una lista (se c'è)

```
void canc (ListeDiElementi *l)
{ if (*l != NULL)
    {ListeDiElementi aus = *l;
    *l = *l -> next;
    free (aus);
    }
}
```



conc(&p);

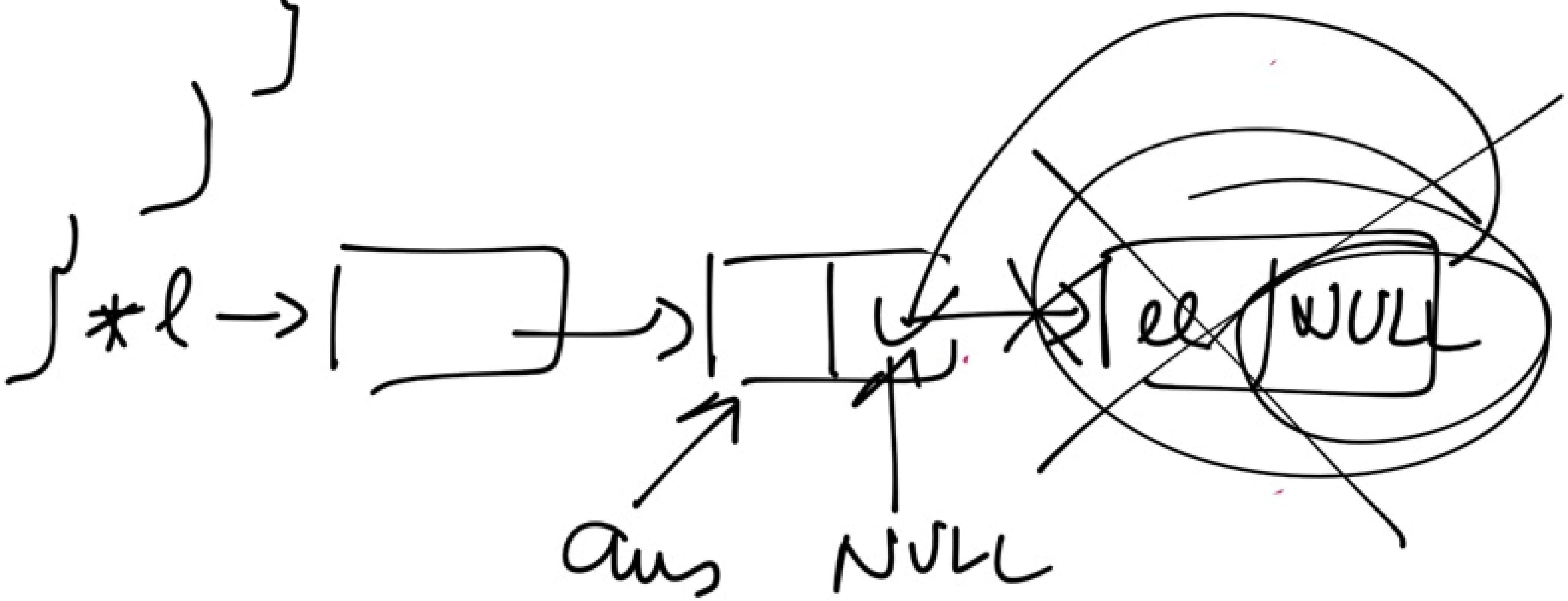
void canc (ListaDiElementi * l ; int el)

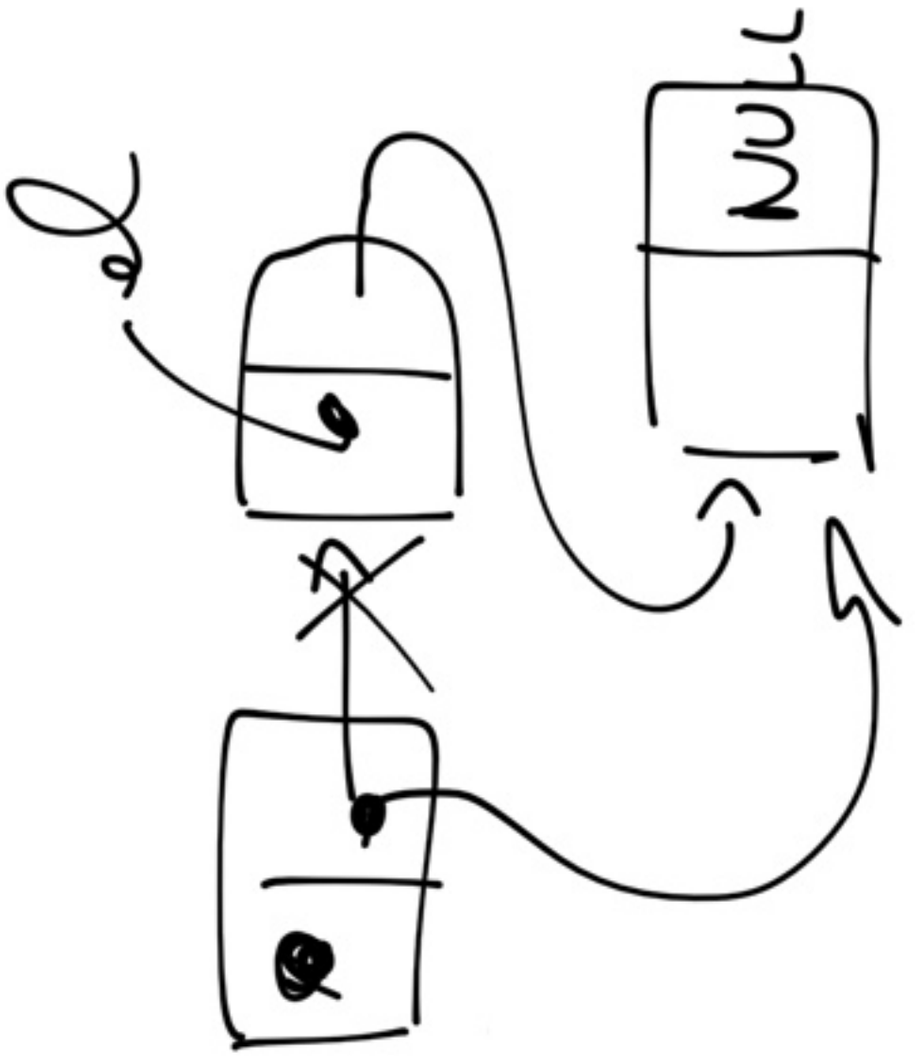


Cancellare il primo
elemento uguale a
el (se c'è)

```
void canc (ListeDiElementi * l ; int el)
{ if (*l != NULL)
  { if (*l->info == el)
      {ListeDiElementi aus; *l = *l->next;
       free (aus); }
    else
      if(*l->next != NULL
      {ListeDiElementi aus = *l; int trovato
                                 =0;
       while (aus->next != NULL && ! trovato)
       if (aus->next->info == el) trovato=1;
       else aus = aus->next;
```

```
if (trovato)
{ ListaDiElementi p = ans->next;
ans->next = ans->next->next;
free(p);
}
}
}
}
```





2013_12_09