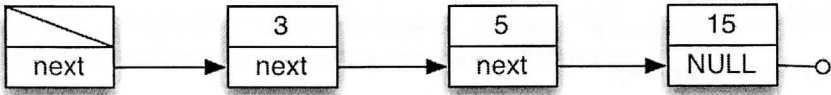


c) funktio, joka lisää uuden int-alkion (**newval**) linkitetyn listan (**l**) loppuun. Käsitteilyn helpottamiseksi linkitettyssä listassa on aina alussa "tyhjä" alkio, joka tulee sivuuttaa (kts. kuva). Älä murehdi muistin vapauttamisesta: se tehdään toisaalla ohjelmassa. Voit myös olettaa, että muistin varaus onnistuu aina. Koodissa on ainakin kaksi virhettä tai puutetta. (2 p)



```
10: #include <stdlib.h>
11:
12: struct list
13: {
14:     int val;
15:     struct list *next;
16: };
17:
18: void add_to_list(struct list *l, int newval)
19: {
20:     if (!l) return;
21:     while (l->next != NULL)
22:         l=l->next;
23:     l->next = malloc(sizeof(int));
24:     l->next->val = newval;
25: }
```

Valgrind palauttaa funktiota suorittaessa seuraavaa:

```
==10788== Invalid read of size 8
==10788==    at 0x400C06: add_to_list (tentti-2014-08-25.c:21)
==10788==    by 0x400FD6: main (tentti-2014-08-25.c:229)
==10788== Address 0x51ba538 is 4 bytes after a block of size 4 alloc'd
==10788==    at 0x4C28BED: malloc (vg_replace_malloc.c:263)
==10788==    by 0x400C18: add_to_list (tentti-2014-08-25.c:23)
==10788==    by 0x400FC2: main (tentti-2014-08-25.c:228)
==10788==
==10788== Invalid write of size 8
==10788==    at 0x400C20: add_to_list (tentti-2014-08-25.c:23)
==10788==    by 0x400FD6: main (tentti-2014-08-25.c:229)
==10788== Address 0x51ba538 is 4 bytes after a block of size 4 alloc'd
==10788==    at 0x4C28BED: malloc (vg_replace_malloc.c:263)
==10788==    by 0x400C18: add_to_list (tentti-2014-08-25.c:23)
==10788==    by 0x400FC2: main (tentti-2014-08-25.c:228)
```

...sekä pari muuta samanlaista ilmoitusta, jotka sivuutettu tilanpuutteen vuoksi. Huomaa että ilmoitusten osoittamilla riveillä ei välttämättä ole virhettä, vaan virhe voi tapahtua esimerkiksi siksi, että jossain toisaalla on jotain väärin.