

Teoretyczne podstawy informatyki
Zestaw zadań nr 8
Wszystkie grupy

dr Anna Ochab-Marcinek

1. Sortowanie szybkie (quicksort). Na czym polega? Jaki jest czas wykonywania algorytmu? Patrz [1].
2. Jak wyglądałoby wykonywanie sumy zbiorów, gdy zbiory reprezentowane są przez dwie nie posortowane listy? Jaki jest czas wykonywania tej operacji? Patrz [2], rozdz. 7.4, listing 7.1.
3. Jak działa funkcja wykonująca sumę zbiorów reprezentowanych przez posortowane listy? Jaki jest czas wykonywania takiej funkcji? Patrz [2], rozdz. 7.4, listing 7.2:

```
#include <stdio.h>

typedef struct CELL *LIST;
struct CELL {
    int element;
    LIST next;
};

LIST setUnion(LIST L, LIST M);
LIST assemble(int x, LIST L, LIST M);

/* assemble produces a list whose head element is x and
   whose tail is the union of lists L and M */

LIST assemble(int x, LIST L, LIST M)
{
    LIST first;

    first = (LIST) malloc(sizeof(struct CELL));
    first->element = x;
    first->next = setUnion(L, M);
}
```

```

        return first;
    }

    /* setUnion returns a list that is the union of L and M */

LIST setUnion(LIST L, LIST M)
{
    if (L == NULL && M == NULL)
        return NULL;
    else if (L == NULL) /* M cannot be NULL here */
        return assemble(M->element, NULL, M->next);
    else if (M == NULL) /* L cannot be NULL here */
        return assemble(L->element, L->next, NULL);
    /* if we reach here, neither L nor M can be NULL */
    else if (L->element == M->element)
        return assemble(L->element, L->next, M->next);
    else if (L->element < M->element)
        return assemble(L->element, L->next, M);
    else /* here, M->element < L->element */
        return assemble(M->element, L, M->next);
}

```

4. Załóżmy, że przed wykonaniem powyższego programu sortujemy obie listy algorytmem quicksort. Jaki jest czas wykonywania go łącznie z sortowaniem? Kiedy jest większy od sumy na nieposortowanych listach, a kiedy mniejszy? Patrz [2], rozdz. 7.4.

Literatura

- [1] http://pl.wikipedia.org/wiki/Sortowanie_szybkie
- [2] A.V. Aho, J.D. Ullman, *Wykłady z informatyki z przykładami w języku C*