Bilaga A
Minnesallokering

Henrik Bäck
850611-6253
Mathias Andersson
850424-6292

Operating Systems DAVB01
VT 2005

Handledare: **Nils Dåverhög**
Hans Hedbom
Thijs Jan Holleboom

Inlämnad **2007-02-12**

```
#define INCREMENT 50
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <sys/time.h>
#include <assert.h>
#include <unistd.h>

/* returnera tidskillnaden mellan tv0 och tv1 i ms */
int difftod(struct timeval * tv0, struct timeval * tv1){
        return (tv1->tv_sec-tv0->tv_sec)*1000 + (tv1->tv_usec-tv0->tv_usec)/1000;
}

/* Pre: true
   Post: objekt skapad med storlek enligt parameter, objektets alla element
        (eller intelligent urval) satta till något värde, och objektet
        makulerat igen. Inget nyttigt har utförts förutom att skapa objektet för
att kunna ta tid på den.
        0 har returnerats om allt gick bra, -1 annars.
*/
int createbigthing(long int storlek)
{
        int* minnet;
        int* alltAllokerat;
        int i;
        int skrivRange = (storlek*1024*1024)/4;
        int retur = -1;

        //Allokera minne att skriva till
        minnet = malloc(storlek*1024*1024);

        //Lagra undan så att vi kan frigöra
        alltAllokerat = minnet;

        if(alltAllokerat != NULL)
        {
                retur = 0;

                //Skriv till minnet
                for(i=0;i<skrivRange;i++)
                {
                        *(minnet++)=1;
                }

        }


        //Avallokera minnet
        free(alltAllokerat);

        minnet = 0;
        alltAllokerat = 0;

        return retur;

}
```

```
int main()
{
        long int i;
        int rtn;
        struct timeval tv0, tv1;
        struct timezone tzp;
        int wentOK = 0;

        /*  Skapa objekt med olika storlekar (i MB) och ta tid*/
        for(i=INCREMENT;i<=1500;i=i+INCREMENT)
        {
                rtn = gettimeofday(&tv0, &tzp); /* set timer T0 */
                wentOK = createbigthing(i);
                rtn = gettimeofday(&tv1, &tzp); /* read time T1 */
                if(wentOK == 0)
                        printf("%d ms | %d MB\n",difftod(&tv0,&tv1),i);
                else
                {
                        printf("Allokering misslyckas på %d MB\n",i);
                        exit(0);
                }
        }
        return 0;
}
```