doit

#! /bin/csh

gcc -o sheet sheet.c
sheet < sheet-test1.in >! output
diff output sheet-test1.out
echo "test over"

foreach i (1 2 3 4)
  echo "iteration $i"
end

Computation structures / concepts:

- Shell scripts
- Execution of Unix commands in shell scripts
- Loops in shell scripts
- Variables in shell scripts
- Execution privilege (chmod Unix command)

layout.c

/* layout.c

This program declares lots of variables of different sizes to illustrate their layout in memory.
*/

#include <stdio.h>
short int a = 1;
short int b = 2;
int c = 3;
int d = 0xffffffff;
int e = 0x7598;
long long int f = 0xaaaaaaaaabbbbbbb;
float g = 3.14159;
float h = 1.41421;
double i = 2.0;
int j[10] = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9};

int main()
{
    int k;

    printf("This program declares lots of variables\n");
}

Computation structures / concepts:

- Declaration of many variables of different size
- gcc -c layout.c (compiles without linking)
- od -A x -X layout.o (dumps compiled module)
- Layout of variables in dump
- Address of variables in dump
- Address of array in dump
- Actual addresses will change

3  pointer.c

/*
   pointer.c

   Illustrates declaration and use of pointers.
*/

#include <stdio.h>
int main(void)
{
    // declaration and initialization of pointer
    int a = 100, b = 200, *p = &a, *q;
    int answer;

    printf("a is stored at memory location %p\n", &a);
    printf("b is stored at memory location %p\n", &b);
    printf("p points to memory location %p\n", p);
    printf("p is stored at memory location %p\n", &p);

    printf("a = %d\n", a);
    printf("*p = %d\n", *p);

    answer = a + 500;
    printf("answer: %d\n", answer);

    answer = *p + 500;
    printf("answer: %d\n", answer);
    p = &b;
    answer = *p + 500;
    printf("answer: %d\n", answer);

    p = &a;
    printf("the value of a is %d\n", a);
    *p = b + *p;
    printf("the value of a after *p = b + *p is %d\n", a);
    printf("the value of *p is %d\n", *p);

    q = p;
    *q = *q + 200;
    printf("the value of a after *q = *q + 200 is %d\n", a);
    printf("the value of *q is %d\n", *q);

    return 0;
}

Computation structures / concepts:

- Address of variables a, b, and p
- Indirection through p
- Changing p changes result of computation
- Storing indirectly through p modifies a