#include <stdio.h>
#include <stdlib.h>

#define MAX 8

struct student
{
    char name[12];
    short exam1;
    short exam2;
    struct student *next;
};

void print_all( struct student * );

int main()
{
    struct student *item, *head, *tail;
    int i;

    head = (struct student *) malloc( sizeof( struct student ) );
    head->next = NULL;

    tail = head;

    for (i=0; i<MAX; i++)
    {
        item = (struct student *) malloc( sizeof( struct student ) );
        item->next = NULL;

        scanf( "%s %hd %hd", &item->name[0], &item->exam1, &item->exam2 );

        tail->next = item;
        tail = item;
    }

    print_all( head->next );
}

<2 lemon:˜/Examples > gcc example26.driver.c example26.support.s

<3 lemon:˜/Examples > a.out < example26.data

<table>
<thead>
<tr>
<th>Current</th>
<th>Student name</th>
<th>Exam 1</th>
<th>Exam 2</th>
<th>Next</th>
</tr>
</thead>
<tbody>
<tr>
<td>001c6020</td>
<td>Brown</td>
<td>70</td>
<td>75</td>
<td>001c6038</td>
</tr>
<tr>
<td>001c6038</td>
<td>Doe</td>
<td>74</td>
<td>84</td>
<td>001c6050</td>
</tr>
<tr>
<td>001c6050</td>
<td>Evans</td>
<td>49</td>
<td>57</td>
<td>001c6068</td>
</tr>
<tr>
<td>001c6068</td>
<td>Fernandez</td>
<td>89</td>
<td>72</td>
<td>001c6080</td>
</tr>
<tr>
<td>001c6080</td>
<td>Gray</td>
<td>66</td>
<td>77</td>
<td>001c6098</td>
</tr>
<tr>
<td>001c6098</td>
<td>Green</td>
<td>85</td>
<td>92</td>
<td>001c60b0</td>
</tr>
<tr>
<td>001c60b0</td>
<td>Jones</td>
<td>81</td>
<td>83</td>
<td>001c60c8</td>
</tr>
<tr>
<td>001c60c8</td>
<td>Smith</td>
<td>50</td>
<td>62</td>
<td>00000000</td>
</tr>
</tbody>
</table>
NULL = 0

.global print_all
.text
.balign 4
print_all:
push {r4,lr}
mov r4, r0          @ 1st argument: address of first record
ldr r0, =fmt1       @ print the column headers
bl printf

loop:
cmp r4, #NULL       @ compare the current address and NULL
beq endloop

ldr r0, =fmt2       @ print first part of one record
mov r1, r4          @ address of the current record
add r2, r4, #0      @ address of name field
bl printf

ldr r0, =fmt3       @ print second part of one record
ldrh r1, [r4, #+12]  @ contents of exam1 field
ldrh r2, [r4, #+14]  @ contents of exam2 field
ldr r3, [r4, #+16]  @ address of the next record
bl printf

ldr r4, [r4, #+16]  @ retrieve the address of the next record
b loop

endloop:
ldr r0, =fmt4       @ print a blank line
bl printf
pop {r4,lr}
bx lr

fmt1: .asci "\n"
.asci "Current  Student name  Exam 1  Exam 2  Next  \n"
.asci "--------  ------------  ------  ------  --------
"

fmt2: .asciz "%08x %-12s "
fmt3: .asciz "%6d %6d %08x\n"
fmt4: .asciz "\n".balign 4