

# EXAMEN INF-154

Wenceslao Palma, Laura Griffiths

1. (30 ptos.) Escriba la función **void eliminarRepetidos(struct nodo \*\*);** la cual elimina todos los nodos cuyo contenido se encuentra repetido en una lista enlazada. Utilice la sgte definición:

```
struct nodo{
    int dato;
    struct nodo *sgte;
};
struct nodo *inicio; // puntero al inicio de la lista.
```

- (a) uso funciones (5 ptos)  
(b) uso punteros (5 ptos)  
(c) solucion (20 ptos)

```
void eliminarRepetidos(struct nodo **inicio){
    struct nodo *i,*j;

    i=*inicio;

    while (i!=NULL){
        antj=i;
        j=i->sgte;
        eliminado=0;
        while (j!=NULL){

            if (i->dato==j->dato){
                antj->sgte=j->sgte;
                free(j);
                j=antj->sgte;
                eliminado=1;
            }else{
                antj=j;
                j=j->sgte;
            }
        }
        if (eliminado){
            if (i==*inicio){
                *inicio=*inicio->sgte;
                free(i);
                i=*inicio;
            }else{
                anti->sgte=i->sgte;
                free(i);
                i=anti->sgte;
            }
        }
        }else{
            anti=i;
            i=i->sgte;
        }
    }
}
```

2. (30 ptos.) Considere una secuencia arbitraria de números enteros. Es posible ubicar los operadores + y - entre los números de la secuencia, de este modo se obtienen diferentes expresiones aritméticas que producen diferentes valores. Por ejemplo para la secuencia: 17, 5, -21, 15 existen 8 posibles expresiones:

```
17 + 5 + -21 + 15 = 16
17 + 5 + -21 - 15 = -14
17 + 5 - -21 + 15 = 58
17 + 5 - -21 - 15 = 28
17 - 5 + -21 + 15 = 6
17 - 5 + -21 - 15 = -24
17 - 5 - -21 + 15 = 48
17 - 5 - -21 - 15 = 18
```

una expresión se dice divisible por K si los operadores + y - son ubicados de tal manera que dan origen a un valor divisible por K. Escriba un programa que determine la divisibilidad de una expresión. Los datos de entrada deben ser una secuencia de N enteros ( $1 \leq N \leq 1000$ ) y K ( $2 \leq K \leq 100$ ). La salida debe ser: La expresión xxxxxx es Divisible por K o ninguna expresión generada es divisible por K.

Considere que tanto los números que componen la secuencia y el valor de K se encuentran almacenados, uno por línea, en un archivo de texto llamado "in.txt". El último número almacenado en "in.txt" corresponde al valor de K. Por ejemplo:

Entrada:

```
17
5
-21
15
3
```

Salida:

```
La expresión 17 - 5 + -21 + 15 = 6 es divisible por 3
La expresión 17 - 5 + -21 - 15 = -24 es divisible por 3
La expresión 17 - 5 - -21 + 15 = 48 es divisible por 3
La expresión 17 - 5 - -21 - 15 = 18 es divisible por 3
```

- (a) lectura archivo (10 ptos)
- (b) generar expresiones (10 ptos)
- (c) verificar divisibilidad (10 ptos)

```
#include <stdio.h>
#include <math.h>

main(){
    int i,m,v,valor_exp,exp,bin;
    int secuencia[1000];
    int k;

    fp=fopen("in.txt","r");

    m=0;
    while (!feof(fp)){
        fscanf(fp,"%d",&v);
        secuencia[m]=v;
        m++;
    }
    fclose(fp);

    for(exp=0;exp<pow(2,m-1);exp++){
        printf("%d",secuencia[0]);
        valor_exp=secuencia[0];
        bin = exp;
        for(j=1;j<=m-1;j++){
            if (bin%2){
                valor_exp -= secuencia[j];
                printf("-%d",secuencia[j]);
            }else{
                valor_exp += secuencia[j];
                printf("+%d",secuencia[j]);
            }
            bin/=2;
        }
        printf("=%d ",valor_exp);
        if (valor_exp%k==0)
            printf(" es divisible por %d\n",k);
        else
            printf(" no es divisible por %d\n",k);
    }
}
```