

## SOLUCIONES ACTIVIDAD 4NP

1) Realiza el diseño de un programa que permita determinar todos los números perfectos entre dos números enteros  $n1$  y  $n2$ . Se debe cumplir que  $n1 < n2$  y, además, que tanto  $n1$  como  $n2$  sean mayores que 0. Definimos los números perfectos del siguiente modo:

- 1 es perfecto
- if  $N > 1$ ,  $N$  es perfecto si la suma de sus divisores (incluido el 1 y excluido el propio número) da como resultado el mismo número.

### ACLARACIONES:

- El programa debe calcular y mostrar la cantidad de números perfectos dentro del intervalo además de visualizar cuáles son.
- En el caso de que  $n1 > n2$  se intercambiarán los valores.
- En el caso de que  $n1$  o  $n2$  sean negativos o cero se repetirá la entrada de datos hasta que sean valores correctos.
- La solución debe construirse empleando módulos.

```
#include <iostream>
using namespace std;

void leerNum(int &n1, int &n2);
void intercambiaValores(int &n1, int &n2);
bool esPerfecto(int num);

int main(){
    int n1, n2, cuenta, i;

    cuenta=0;
    leerNum(n1,n2);
    if (n1==1) {
        cout << "el número " << 1 << " es perfecto" << endl;
        cuenta++;
    }
    for (i=n1; i<=n2; i++){
        if (esPerfecto(i)){
            cout << "el número " << i << " es perfecto" << endl;
            cuenta=cuenta+1;
        }
    }
    cout << "Hay " << cuenta << " números perfectos" << endl;
}

void leerNum(int &n1, int &n2){
    do{
        cout << "Introduce un número ";
        cin >> n1;
        cout << "Introduce otro número mayor ";
        cin >> n2;
    }while (n1<=0 || n2<=0 || n1==n2);
    if (n1>n2)
        intercambiaValores(n1,n2);
}
```

```

void intercambiaValores(int &n1, int &n2){
    int aux;

    aux=n1;
    n1=n2;
    n2=aux;
}

```

```

bool esPerfecto(int num){
    int suma, i;
    bool res;

    suma=0;
    for (i=1; i<=num/2; i++)
        if (num%i==0) //i es divisor de num
            suma=suma+i;
    if (suma==num)
        res=true;
    else
        res=false;
    return (res);
}

```

- 2) } Implementa un programa que lea un número y un carácter desde teclado. El número debe estar comprendido entre 10 y 100. Si el carácter es una letra minúscula debe imprimir en pantalla el carácter correspondiente en mayúsculas tantas veces como indique el número leído, si el carácter es una letra mayúscula debe imprimir en pantalla el carácter correspondiente en minúsculas tantas veces como indique el número leído. No se usará la letra 'ñ'. El programa debe estar compuesto por varios módulos.

```

#include <iostream>
#include <ctype.h>
using namespace std;

void leeDatos(int &num, char &car);
void procesa(int num, char car);
void muestra(int num, char car);

int main(){
    int n;
    char car;

    leeDatos(n, car);
    procesa(n, car);
}

void leeDatos(int &num, char &car){
    do{
        cout << "Introduce un número ";
        cin >> num;
        if (num<10 || num>100)
            cout << "EL NUMERO TIENE QUE ESTAR ENTRE 10 Y 100" << endl;
    }while(num<10 || num>100);

    do {
        cout << "Introduce un caracter ";
        cin >> car;
        if (!((car >= 'a' && car <= 'z') || (car >= 'A' && car <= 'Z'))
            cout << "CARÁCTER NO VÁLIDO, DEBE SER UNA LETRA " << endl;
    } while (!((car >= 'a' && car <= 'z') || (car >= 'A' && car <= 'Z')));
}

```

```
void procesa(int num, char car){
    if (car>='A' && car<='Z')
        muestra(num, car+32);
    else
        muestra(num, car-32);
}

void muestra(int num, char car){
    int i;

    for (i=1; i<=num; i++)
        cout << car;
    cout << endl;
}
```