

```

1: //////////////////////////////////////////////////
2: //
3: // Fundamentos de Programación
4: // ETS Informática y Telecomunicaciones
5: // Universidad de Granada
6: // // Don Oreó
7: //
8: //////////////////////////////////////////////////
9:
10: #include <iostream>
11: using namespace std;
12:
13: int main() {
14:     /*
15:     Esquema alumnos:
16:
17:         Filtro para la distancia d (debe ser d >= 0):
18:
19:         do{
20:             cin >> d;
21:         }while (d < 0);
22:
23:         Filtro para el número de ptos n:
24:
25:         do{
26:             cin >> n;
27:         }while (n < MIN || MAX < n);
28:     */
29:
30:     const int LIM_KMS_RECARGO = 300;
31:     const double RECARGO_KM = 0.1;
32:     const double MIN_KMS_DSCTO = 700.0;
33:     const int MIN_PTOS_DSCTO_BAJO = 100;
34:     const int MIN_PTOS_DSCTO_ALTO = 200;
35:     const int DSCTO_KMS = 2;
36:     const int DSCTO_BAJO_PTOS = 3;
37:     const int DSCTO_ALTO_PTOS = 4;
38:     const double TRF_BASE = 150.0;
39:     const int MAX_NUM_PTOS = 400;
40:     const int MIN_NUM_PTOS = 0;
41:     const char NUEVO_BILLETE='N';
42:     const char TERMINADOR='#';
43:
44:     double dscto;
45:     double trf = 0;
46:     int dist;
47:     int ptos_fideliz;
48:     char billete;
49:
50:     //Damos por hecho que minimo vas a comprar 1 billete
51:
52:     cout << "Tarifa aerea."
53:         << "\nIntroduzca 'N' para un nuevo billete o '#' para finalizar la compra:" << endl;
54:
55:     while(billete!=NUEVO_BILLETE && billete!=TERMINADOR)
56:         cin >> billete;
57:
58:     while(billete!=TERMINADOR) {
59:
60:         cout << "\nIntroduzca la distancia del recorrido del viaje (> 0) y el "
61:             << "número de puntos de la tarjeta de fidelización (entre 0 y "
62:             << MAX_NUM_PTOS << ":\n";
63:
64:         do{
65:             cin >> dist;
66:         }while (dist < 0);
67:
68:         do{
69:             cin >> ptos_fideliz;
70:         }while (ptos_fideliz < MIN_NUM_PTOS
71:             ||
72:             MAX_NUM_PTOS < ptos_fideliz);
73:
74:
75:         /*
76:         Algoritmo:
77:             Inicializar la tarifa a la tarifa base
78:
79:             Según sea la longitud del trayecto
80:                 Actualizar la tarifa
81:
82:             Según sea la longitud del trayecto
83:                 Inicializar el dscto
84:
85:             Según sea el número de puntos
86:                 Actualizar el dscto

```

```
87:
88:     Aplicar el dscto calculado anteriormente a la tarifa
89:     */
90:
91:
92:     trf = TRF_BASE;
93:
94:     if (dist > LIM_KMS_RECARGO)
95:         trf = trf + RECARGO_KM*(dist - LIM_KMS_RECARGO) ;
96:
97:
98:     if (dist > MIN_KMS_DSCTO)
99:         dscto = DSCTO_KMS;
100:    else
101:        dscto = 0;
102:
103:    if (ptos_fideliz > MIN_PTOS_DSCTO_ALTO)
104:        dscto = dscto + DSCTO_ALTO_PTOS;
105:    else if (ptos_fideliz > MIN_PTOS_DSCTO_BAJO)
106:        dscto = dscto + DSCTO_BAJO_PTOS;
107:
108:    trf = trf * (1 - dscto / 100.0);
109:
110:    cout << "\n\nTarifa final aplicando los dsctos: ";
111:    cout << trf;
112:
113:    billete=0;
114:    cout << "\n-----" << endl;
115:    cout << "\nIntroduzca 'N' para un nuevo billete o '#' para finalizar la compra:" << endl;
116:    while(billete!=NUEVO_BILLETE && billete!=TERMINADOR){
117:        cin >> billete;
118:    }
119: }
120: }
```

```
1: //////////////////////////////////////////////////////////////////////
2: //
3: // Fundamentos de Programación
4: // ETS Informática y Telecomunicaciones
5: // Universidad de Granada
6: // // Don Oreó
7: //
8: //////////////////////////////////////////////////////////////////////
9:
10: #include <iostream>
11:
12: using namespace std;
13:
14: int main() {
15:     int min;
16:     int max;
17:     int k;
18:     int divisor;
19:     double tope;
20:
21:     divisor = 0;
22:     k = 0;
23:     max = 0;
24:
25:     //Entrada de Datos...
26:
27:     cout << "Introduce un numero minimo: ";
28:     cin >> min;
29:
30:     cout << "Introduce un numero maximo: ";
31:     while (max<min)
32:         cin >> max;
33:
34:     cout <<"Introduce el limite de divisores: ";
35:     while (k<1)
36:         cin >> k;
37:
38:     //Cómputo...
39:
40:     for(int i=min; i<=max; i++) {
41:         tope=i/2;
42:         for(int j=2; j<tope; j++) {
43:             if(i%j==0)
44:                 divisor++;
45:         }
46:         if(divisor>=k)
47:             cout << i << " ";
48:
49:         divisor=0;
50:     }
51:
52: }
53:
54:
```

```

1: //////////////////////////////////////
2: //
3: // Fundamentos de Programación
4: // ETS Informática y Telecomunicaciones
5: // Universidad de Granada
6: // // Don Oreó
7: //
8: //////////////////////////////////////
9: /*
10: 3. [Sustituir carácter por vector (con vector auxiliar)] Tenga en cuenta la observación al
11: inicio de esta relación de problemas sobre la lectura de los caracteres (ver página RPIII.1).
12: Para poder leer caracteres, incluyendo los espacios en blanco, hay que usar caracter = cin.get(),
13: en vez de cin >> caracter.
14:
15: Dado un vector de caracteres, queremos sustituir todas las apariciones de un carácter
16: y poner en su lugar el contenido de otro vector.
17:
18: Por ejemplo, si tenemos el vector [u n o a d o s a], el resultado de
19: sustituir las apariciones del carácter 'a' por el nuevo vector [T T U] sería
20: [u n o T T U d o s T T U T T U]
21:
22: Resolveremos este problema de varias formas a lo largo de esta Relación de Problemas.
23: En este ejercicio, se construirá un tercer vector sustituido con el resultado
24: pedido.
25:
26: Construya un programa que lea caracteres hasta que se introduzca # lo que formará
27: el primer vector (v). A continuación lea el carácter a_borrar que se va a eliminar de
28: v. Finalmente, el programa leerá los caracteres que formarán el vector a_insertar
29: que sustituirán cada aparición de a_borrar. El terminador de entrada de caracteres
30: para el vector a_insertar es también el carácter #
31:
32: El programa construirá e imprimirá en pantalla un tercer vector sustituido que
33: contendrá los caracteres de v pero reemplazando todas las apariciones de a_borrar
34: por los caracteres del vector a_insertar. Si el vector a_insertar contuviese el
35: carácter a_borrar, dichas apariciones no se eliminan, tal y como puede apreciarse
36: en el último ejemplo que aparece al final de este enunciado.
37:
38: Para realizar la tarea pedida, se recomienda que implemente el siguiente algoritmo:
39:
40: Recorrer las componentes -i- del vector v
41:     Si v[i] == a_borrar
42:         Añadir a sustituido todas las componentes del vector a_insertar
43:     si no
44:         Añadir a sustituido la componente v[i]
45:
46: Puede utilizar el esbozo del programa disponible en el siguiente enlace:
47: http://decsai.ugr.es/jccubero/FP/III\_SustituyeCaracterVectorEsbozo.cpp
48:
49: Ejemplo de entrada: u n o a d o s a a # T T U # a
50: -- Salida correcta: u n o T T U d o s T T U T T U
51: Ejemplo de entrada: u n o a d o s a a # T a U # a
52: -- Salida correcta: u n o T a U d o s T a U T a U
53: Finalidad: Trabajar con vectores auxiliares. Dificultad Baja.
54: */
55: #include <iostream>
56: #include <cmath>
57:
58: using namespace std;
59:
60: int main(){
61:     int k=0;
62:     int util_v = 0;
63:     int util_a_insertar = 0;
64:     int util_total;
65:     const int TAMANIO=100;
66:     char v[TAMANIO];
67:     char a_insertar[TAMANIO];
68:     char sustituido[TAMANIO];
69:     char caracter;
70:     char caracter_a_borrar;
71:     const char TERMINADOR = '#';
72:
73:     //Entrada de Datos...
74:
75:     cout << "Introduce el vector inicial: ";
76:     caracter = cin.get();
77:
78:     while (caracter != TERMINADOR && util_v < TAMANIO){
79:         v[util_v] = caracter;
80:         util_v++;
81:         caracter = cin.get();
82:     }
83:     cout << "Introduce el caracter a borrar: ";
84:     cin >> caracter_a_borrar;
85:     caracter = cin.get();
86:

```

```
87:     caracter=0;    //o cualquier otra cosa diferente a TERMINADOR
88:
89:     cout << "Introduce el vector a insertar: ";
90:     caracter = cin.get();
91:
92:     while (caracter != TERMINADOR && util_a_insertar < TAMANIO){
93:         a_insertar[util_a_insertar] = caracter;
94:         util_a_insertar++;
95:         caracter = cin.get();
96:     }
97:
98: //Cómputo...
99:
100:    util_total=util_v;
101:
102:    for(int i=0;i<util_total;i++){
103:        if (v[i]!=caracter_a_borrar){
104:            sustituido[k]=v[i];
105:            k++;
106:        }
107:        else{
108:            util_total+=util_a_insertar-1;
109:            for(int j=0;j<util_a_insertar;j++){
110:                sustituido[k]=a_insertar[j];
111:                k++;
112:            }
113:        }
114:    }
115: //Salida de Datos...
116:
117:    for (int i = 0; i < util_total; i++)
118:        cout << sustituido[i];
119:
120:
121:
122:
123: }
124:
```

```
1: //////////////////////////////////////
2: //
3: // Fundamentos de Programación
4: // ETS Informática y Telecomunicaciones
5: // Universidad de Granada
6: // // Don Oreó
7: //
8: //////////////////////////////////////
9:
10: #include <iostream>
11: using namespace std;
12:
13: int main() {
14:     int computo=0,util=0,i=0,j=0,solucion;
15:     int pos=0; /*Inicializamos a 0 para comparar al final si no ha sido modificado e imprimir que "no ha
y solucion" */
16:
17:     //Introduccion de datos
18:     cout << "Introduce el valor de util: ";
19:     cin >> util;
20:     int v[util];
21:
22:     cout << "Introduce el vector: ";
23:     for (i=0;i<util;i++){
24:         cin >> v[i];
25:     }
26:     i=0;
27:
28:     cout << "Inserte el solucion: ";
29:     cin >> solucion;
30:
31:     //Cómputo
32:     for (i=0;i<util;i++){
33:         for (j=i;j<util;j++){
34:             computo=computo+v[j];
35:             if(computo == solucion){
36:                 pos=i;
37:             }
38:         }
39:         computo=0;
40:     }
41:     if(pos>0)
42:         cout << pos+1 << endl; /*El mas 1 es para que la posicion a imprimir sea logica*/
43:     else
44:         cout << "No hay solucion" << endl;
45: }
```

```
1: //////////////////////////////////////
2: //
3: // Fundamentos de Programación
4: // ETS Informática y Telecomunicaciones
5: // Universidad de Granada
6: // Departamento de Ciencias de la Computación e Inteligencia Artificial
7: // Autor: Don Oreo
8: //
9: //////////////////////////////////////
10:
11: // Frecuencias
12: /*
13:     Algoritmo:
14:
15:     Recorrer -i- el vector texto
16:         actual = texto[i]
17:
18:     Si actual no está en el vector procesados:
19:         - Añadir actual a procesados
20:         - Contar el número de ocurrencias de actual
21:           en el vector texto -a partir de la posición i+1-
22: */
23: #include <iostream>
24: using namespace std;
25:
26: int main(){
27:     const char TERMINADOR = '@';
28:     const int NUM_CARACT_ASCII = 256;
29:     const int MAX_NUM_CARACT = 1e4; // diez mil
30:
31:     char a_buscar[NUM_CARACT_ASCII];
32:     int frecuencias[NUM_CARACT_ASCII] {};
33:     char texto[MAX_NUM_CARACT];
34:     int contador;
35:     char car;
36:     int util_a_buscar, util_texto;
37:
38:     cout << "Frecuencias\n"
39:          << "Introduzca los caracteres del vector a buscar, con terminador "
40:          << TERMINADOR << "\n"
41:          << "A continuación introduzca los caracteres del texto,"
42:          << " usando el mismo terminador.\n\n";
43:
44:     // Introduccion de Datos
45:
46:     car = cin.get();
47:     util_a_buscar = 0;
48:
49:     while (car != TERMINADOR && util_a_buscar < NUM_CARACT_ASCII){
50:         a_buscar[util_a_buscar] = car;
51:         car = cin.get();
52:         util_a_buscar++;
53:     }
54:
55:     car = cin.get();
56:     util_texto = 0;
57:
58:     while (car != TERMINADOR && util_texto < MAX_NUM_CARACT){
59:         texto[util_texto] = car;
60:         car = cin.get();
61:         util_texto++;
62:     }
63:
64:     //Computo de Datos
65:
66:     /*//Metodo 1:
67:     for(int i=0;i<NUM_CARACT_ASCII;i++){
68:         car=i;
69:         for(int j=0;j<util_texto;j++){
70:             if(car==texto[j])
71:                 frecuencias[i]++;
72:         }
73:     }
74:
75:     //Salida de Datos
76:
77:     for(int j=0;j<util_a_buscar;j++)
78:         for(int i=0;i<NUM_CARACT_ASCII;i++)
79:             if(a_buscar[j]==i)
80:                 cout << i << " : " << frecuencias[i] << endl;
81:
82:     */
83:     //Metodo 2:
84:
85:     for(int j=0;j<util_a_buscar;j++)
86:         for(int i=0;i<util_texto;i++)
```

```
87:         if(a_buscar[j]==texto[i])
88:             frecuencias[j]++;
89:
90:     //Salida de Datos
91:
92:     for(int j=0;j<util_a_buscar;j++)
93:         cout << j << " : " << frecuencias[j] << endl;
94:
95:     /*
96:     Ja@Juan Carlos Cubero@°
97:
98:     J: 1
99:     a: 2
100:    */
101: }
102:
103:
```