tecnun
CAMPUS TECNOLÓGICO DE LA UNIVERSIDAD DE NAVARRA
NAFARROAKO UNIBERTSITATEKO CAMPUS TEKNOLOGIKOA
Escuela Superior de Ingenieros · Ingeniarien Goi Mailako Eskola

***Carnet*: _____** ***Nombre*: _____**

C++

Practica Calificada 19-20

Informática II

Dr. Paul Bustamante /Dr. Ibon Elosegui

ÍNDICE

1. Introducción.

Recuerde que si no entiende bien el enunciado de los ejercicios o no está seguro de lo que se le pide, puede preguntar.

1.1 Simulación Caída Libre (3.0)

Este ejercicio consiste en calcular la posición de un cuerpo en caída libre. La ecuación que rige este movimiento es:

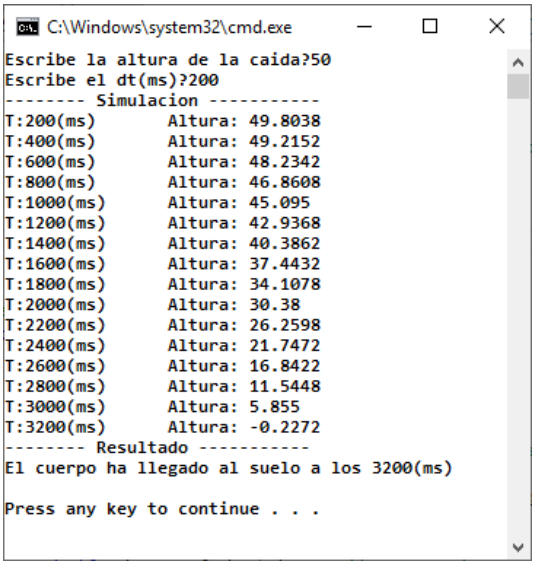
$$y = h - \frac{g * t^2}{2.0}$$ h: *altura inicial (mts.)* g: *9.81m/s^2*

Se pide sacar por la consola las distintas posiciones del cuerpo para cada diferencial de tiempo (***dt***), hasta que el cuerpo llegue a la superficie. Se debe especificar además, el tiempo en el que el cuerpo ha llegado al suelo en (**ms**).

Los datos que el usuario debe dar son dos:

1) Altura de la caída del cuerpo (en mts.).
2) Diferencial de tiempo para el cálculo de las distintas posiciones(en milisegundos).

Para realizar este ejercicio, deberá hacer la función `double Calcula(double alt, double dt)`, la cual hará la simulación (ver la figura adjunta), sanado por consola la altura en cada instante "**dt**". Esta función deberá devolver el tiempo al cual el cuerpo ha llegado al suelo (y=0). Como se puede ver en la figura, dicho tiempo será más exacto cuando el "**dt**" sea más pequeño.

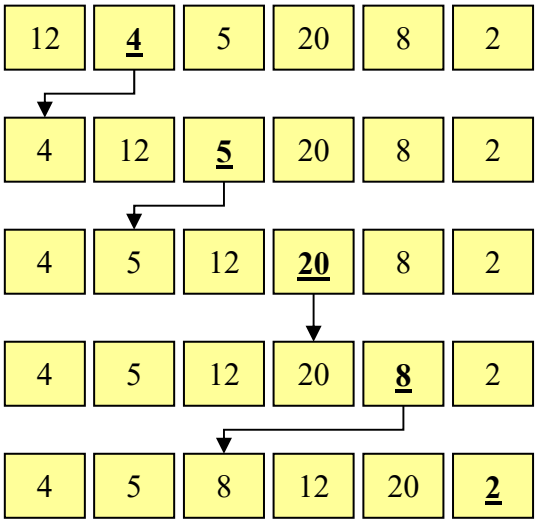


1.2 Ordenación por Inserción (4.0)

El algoritmo de ordenación por inserción es más eficiente que el algoritmo de la burbuja. Si los datos están casi ordenados, necesitará **menos** iteraciones.

Este algoritmo consiste en lo siguiente: Inicialmente se ordenan los 2 primeros elementos, luego se inserta el tercero en la posición correcta con respecto a los 2 primeros que ya están ordenados, después el cuarto elemento se inserta en la posición adecuada con respecto a los tres primeros elementos ya ordenados y así sucesivamente hasta el final.

En el ejercicio debe pedir los números a ordenar, usando **reserva dinámica** de memoria, y finalmente, se debe sacar por consola tanto el número total de iteraciones realizadas, como el vector que contiene a todos los elementos ordenados



Para realizar este ejercicio, deberá realizar las siguientes funciones:

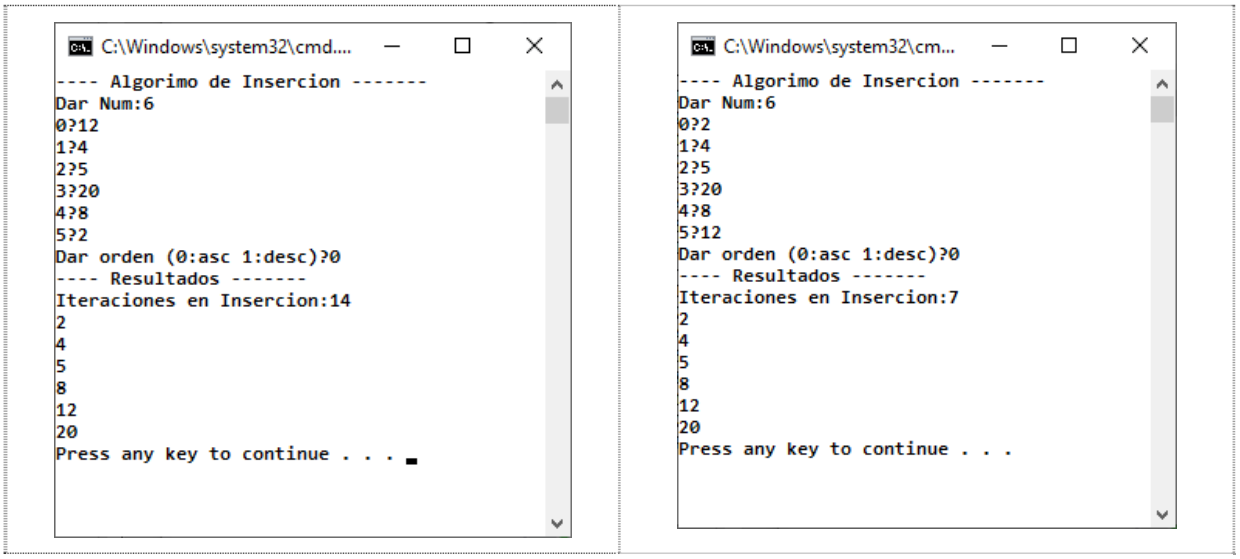
1. `int *Insercion( int *vec , int n, bool orden)`
2. `void print(int *vec, int n);`
3. `int *pidedatos(int &n);`

La función **Inserción**, recibirá el vector con los datos desordenados y deberá devolver otro vector (usando reserva dinámica de memoria) con los datos ordenados, según la variable "**orden**", la cual los ordenará de forma ascendente o descendente. Además, esta función deberá imprimir el **número de iteraciones** que realiza para hacer el algoritmo de ordenación.

La función **pidedatos** deberá devolver el vector con los datos (reserva dinámica de memoria).

La función **print** se encargará de imprimir el vector que se le pase. En todo caso "n" es el número de elementos que hay en el vector.

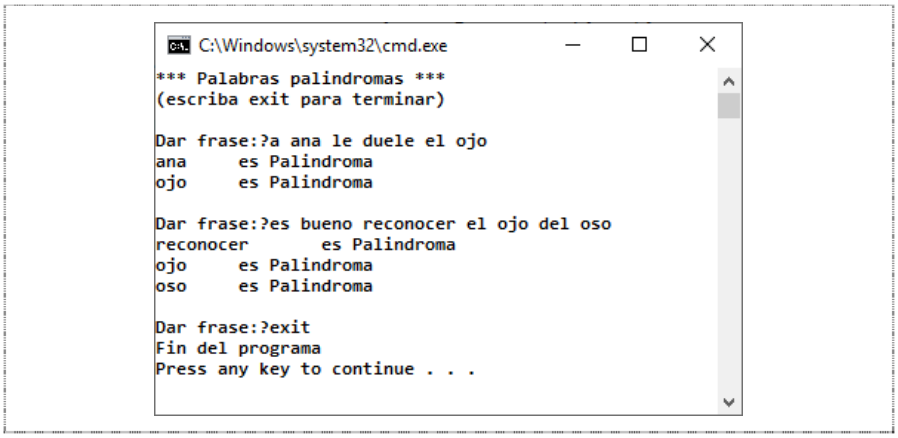
En las siguientes figuras puede ver que, si el vector está medianamente ordenado, hará menos iteraciones, lo cual no ocurre con el método de la burbuja, por ejemplo.



1.3 Palabras Palíndromas *(3.0)*

Una frase o palabra palíndroma es aquella que se lee igual en un sentido que en otro (por ejemplo Ana, Oso, Ojo).

Este ejercicio consiste en hacer un programa que pida una frase e imprima en pantalla todas las palabras **palíndromas** que encuentre (vea el ejemplo de la figura siguiente).



Para realizar el ejercicio, deberá tener en cuenta las siguientes condiciones:

- El programa siempre estará corriendo hasta que el usuario escriba "**exit**".
- Deberá hacer **dos** funciones:
  - La función `bool palindroma(char *palabra)`, a la cual se le pasará una palabra y dirá si es palíndroma o no.
  - La función `char *pideFrase()` , la cual devolverá la **frase escrita** por el usuario (usando reserva dinámica de memoria). En main deberá sacar la palabras de la frase escrita y usar la **funcion palindroma**.

Buena suerte a todos!!!