Francisco José Caballero del Campo

Imagen de la pantalla de un video juego

Descripción generada automáticamente con confianza media

Memoria 2

Traslado de piezas en bloque

Tabla de contenido

[Versión básica 2](#_Toc104740208)

[Estación de partida 2](#_Toc104740209)

[Ejercicio básico 2](#_Toc104740210)

[Simulaciones 4](#_Toc104740211)

[Simulación con 7 piezas 4](#_Toc104740212)

[Simulación con 4 piezas 4](#_Toc104740213)

[Versión avanzada (I) 6](#_Toc104740214)

[Estación de partida 6](#_Toc104740215)

[Ejercicio Avanzado 1 6](#_Toc104740216)

[Simulaciones 9](#_Toc104740217)

[Simulación con 10 piezas y 4 torres con radio 300 9](#_Toc104740218)

[Simulación con 17 piezas y 3 torres con radio 200 9](#_Toc104740219)

[Simulación 10 piezas y 4 torres con radio 300 10](#_Toc104740220)

[Versión Avanzada (II) 11](#_Toc104740221)

[Estación de partida 11](#_Toc104740222)

[Ejercicio avanzado 2 11](#_Toc104740223)

[Simulaciones 13](#_Toc104740224)

[Simulación 3 piezas 13](#_Toc104740225)

# Versión básica

## Estación de partida

Imagen que contiene lego, juguete, foto, diferente

Descripción generada automáticamente

Montaje según lo hecho en clase.

## Ejercicio básico

Para la realización de este ejercicio, debemos llevar la torre del punto origen al punto destino moviendo las piezas de una en una y respetando el orden inicial. Para ellos, necesitamos montar otra torre en un punto intermedio donde las piezas estarán en orden inverso al deseado y desde ese punto intermedio, moverlas de una en una al punto destino, donde tendrá el orden deseado.

El código Rapid realizado para este ejercicio consta de una función llamada MueveTorre cuyos parámetros de entrada son: un punto de origen, un punto destino, el número de piezas de la torre origen y un número para el desplazamiento, este desplazamiento sirve para mover la posición en el eje Y el punto donde se encontrará la torre intermedia, su valor se contará según el número de diámetros que queramos desplazar las torres.

Dicha función esta formada principalmente por dos bucles FOR anidados, el primero de ellos sirve para recorrer cada uno de los puntos de la torre, es decir, recorre una variable llamada “tabPos{3}” con tres puntos dentro, siendo la posición uno el punto origen, la posición 2 el punto intermedio y la posición 3 el punto destino, de esta forma en la primera iteración solo accedemos a los puntos origen e intermedio y en la segunda a los puntos intermedio y destino; y el segundo bucle FOR sirve para recorrer todos los puntos de las piezas que conforman la torre “origen” para llevarlas al “destino” activando y desactivando la ventosa.

El código Rapid es el siguiente:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente con confianza media

## Simulaciones

### Simulación con 7 piezas

Imagen que contiene lego, juguete

Descripción generada automáticamente

Imagen que contiene lego, juguete, tabla, computadora

Descripción generada automáticamente

Imagen que contiene lego, tráfico

Descripción generada automáticamenteImagen que contiene lego, juguete, remoto

Descripción generada automáticamente

Simulación con 7 piezas en la posición origen y un desplazamiento de 4.

### Simulación con 4 piezas

Diagrama

Descripción generada automáticamente con confianza bajaImagen que contiene lego

Descripción generada automáticamente

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamenteImagen que contiene lego, juguete, tabla, luz

Descripción generada automáticamente

Simulación con 4 piezas en la posición origen y desplazamiento 5.

# Versión avanzada (I)

## Estación de partida

Imagen que contiene computadora

Descripción generada automáticamente

Estación de partida dada por el profesor.

Consta de una torre de diez piezas apiladas en una posición origen y un punto destino con círculos concéntricos con radios de 100-200-300-400 milímetros.

## Ejercicio Avanzado 1

Para la realización de este ejercicio tenemos que mover las piezas de la torre origen de tal manera que proporcionando un radio y un numero de torres en el destino, dichas piezas formen torres más pequeñas en el circulo con el radio proporcionado y separadas entre si la misma distancia con una altura equivalente, en el caso de que se pueda.

El código Rapid de este ejercicio consta de una función Ejercicio2 cuyos parámetros de entrada son: numero de piezas en el origen, numero de torres y radio de la circunferencia.

Esta función está formada por tres bucles FOR, en el primero calculamos y guardamos en un “array” los puntos donde se ubicarán las torres en el circulo seleccionado. Cogiendo como referencia el punto destino, calculamos la posición en el eje X haciendo uso del seno y la posición en el eje Y haciendo uso del coseno.

Gráfico, Diagrama, Gráfico radial

Descripción generada automáticamente

Los otros bucles FOR son dos bucles anidados de tal forma que en el primero vamos aumentando la altura de las torres destino cuando estás tengan la misma altura y en el segundo bucle, vamos recorriendo la torre origen y poniendo las piezas en las torres que se encuentran en el círculo, añadimos también una restricción para que en el caso de que se acaben las piezas en el origen y no tengan todas las torres destino la misma altura, el brazo se quede en la posición de reposo.

El código Rapid es el siguiente:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Texto, Aplicación

Descripción generada automáticamente

## Simulaciones

### Simulación con 10 piezas y 4 torres con radio 300

Diagrama

Descripción generada automáticamente con confianza bajaImagen que contiene Flecha

Descripción generada automáticamente

Imagen que contiene electrónica, computadora, cd

Descripción generada automáticamente

Podemos observar que la altura de las torres en el destino es diferente y el brazo acaba en la posición de reposo.

### Simulación con 17 piezas y 3 torres con radio 200

Imagen que contiene Diagrama

Descripción generada automáticamente

Imagen que contiene computadora, tabla

Descripción generada automáticamente

Imagen que contiene computadora

Descripción generada automáticamente

### Simulación 10 piezas y 4 torres con radio 300

Imagen de la pantalla de un video juego

Descripción generada automáticamente con confianza baja

# Versión Avanzada (II)

## Estación de partida

Diagrama, Icono

Descripción generada automáticamente

Estación de partida proporcionada por el profesor.

## Ejercicio avanzado 2

Este ejercicio trata de resolver el problema de las torres de Hanoi, es decir, mover la torre existente de una posición a otra, moviendo las piezas de una en una con la restricción de que no se puede poner una pieza encima de otra más pequeña.

Este ejercicio no he sido capaz de resolverlo ya que me ha sido imposible calculas a que altura se encuentra cada pieza en cada iteración.

En el código Rapid tenemos una llamada a la función TorresHanoi cuyas entradas son: numero de piezas, punto origen, punto destino, punto intermedio y altura de la pieza.

Dentro de la función tenemos una expresión IF-ELSE la cual va llamándose de manera recursiva.

El código Rapid es el siguiente:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Imagen que contiene Texto

Descripción generada automáticamente

Los valores de la altura no tienen mucho sentido, ya que intenté que los puntos que se le pasan como parámetro a la función al hacer la recursión tuvieran la altura de cada una de las torres, pero debido a la recursión no funcionó.

## Simulaciones

### Simulación 3 piezas

Diagrama

Descripción generada automáticamenteImagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

Como podemos observar, no se lleva a cabo la realización del ejercicio.

Activando y desactivando las señales de la ventosa, he podido comprobar que el algoritmo utilizado funciona correctamente, faltaría resolver el problema de las alturas para concluirlo.