Life Line 1

static int BinarySearch(string [] A, int top, int bot, string itemToFind )

{

int mid=0;

return mid;

}

Life Line 2

static int BinarySearch(string [] A, int top, int bot, string itemToFind )

{

int mid = (top + bot) / 2;

if (bot>top)

{

return -1;

}

else if (itemToFind == A[mid])

{

return mid;

}

return 0;

}

Life Line 3

static int BinarySearch(string [] A, int top, int bot, string itemToFind )

{

int mid = (top + bot) / 2;

if (bot>top)

{

return -1;

}

else if (itemToFind == A[mid])

{

return mid;

}

else if (String.Compare(itemToFind, A[mid])==1)

{

return

}

}

Life line 4

class Program

{

static void Main(string[] args)

{

string[] animals = new string[] { "Aardvark", "Ape", "Baboon", "Chaffinch", "Dog", "Elephant", "Fish", "Giraffe", "Horse", "Kitten", "Locust", "Mongoose", "Ostrich", "Parrott", "Rabbit", "Snake", "Tiger", "Vole", "Wolf", "Zebra" };

Console.WriteLine(BinarySearch(animals, animals.Length - 1, 0, "Zebra"));

Console.ReadLine();

}

static int BinarySearch(string [] A, int top, int bot, string itemToFind )

{

int mid = (top + bot) / 2;

if (bot>top)

{

return -1;

}

else if (itemToFind == A[mid])

{

return mid;

}

else if (String.Compare(itemToFind, A[mid])==1)

{

return BinarySearch(A, top, mid + 1, itemToFind);

}

else if (String.Compare(itemToFind, A[mid]) == -1)

{

return BinarySearch(A, mid-1, bot, itemToFind);

}

else

{

return -1;

}

}

}

}

Life Line 5

Animal[] animals;

animals = new Animal[19];

animals[0] = new Animal("Aardvark", 8);

Life Line 6

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApp8

{

class Program

{

static void Main(string[] args)

{

Animal[] animals;

animals = new Animal[19];

animals[0] = new Animal("Aardvark", 8);

animals[1] = new Animal("Ape", 34);

animals[2] = new Animal("Baboon", 45);

animals[3] = new Animal("Chaffinch", 3);

animals[4] = new Animal("Dog", 12);

animals[5] = new Animal("Elephant", 24);

Console.WriteLine(BinarySearch(animals, 5, 0, "Dog"));

Console.ReadLine();

}

static int BinarySearch(Animal [] A, int top, int bot, string itemToFind )

{

int mid = (top + bot) / 2;

if (bot>top)

{

return -1;

}

else if (itemToFind == A[mid].getType())

{

return mid;

}

else if (String.Compare(itemToFind, A[mid].getType()) == 1)

{

return BinarySearch(A, top, mid + 1, itemToFind);

}

else if (String.Compare(itemToFind, A[mid].getType()) == -1)

{

return BinarySearch(A, mid-1, bot, itemToFind);

}

else

{

return -1;

}

}

}

class Animal

{

private string Type;

private int lifeSpan;

public Animal(string t, int l)

{

Type = t;

lifeSpan = l;

}

public string getType()

{

return Type;

}

}

}