דף עזר לבחינה

int grade		הגדרת משתנה

if (grade > 60) {
Console.WriteLine(“Pass”)
}
else
{
Console.WriteLine(“Fail”)
}
		if משפט

switch (a)
{
case 0:
Console.WriteLine("zero");
break;
case 1:
Console.WriteLine("one");
break;
default:
Console.WriteLine("default");
break;
}
		switch משפט

s.EndsWith(“ing”)	→ true	string s = “programming”
s.StartsWith(“pr”)	→ true	
s.IndexOf(“g”, 6)	→ 10	
s.Insert(0, “the “)	→ “the programming”	
s.Remove(1, 2)	→ “pgraming”	
s.Replace(“ing”, ““)	→ “program”	
s.Substring(4, 3)	→ “ram”	
i = Convert.ToInt32(d)	→ i = 7	int i = 5, double d = 7.7, string s = “2”
i = Int32.Parse(s)	→ i = 2	
s = i.ToString()	→ s = “5”	

for (int i = 1; i < 10; i++)
{
Console.WriteLine(i);
}
		for לולאת

while(a < 4)
{
a = a + 1;
}
		while לולאת

int[,] integerArray = new int[10,5];		הגדרת מערך

integerArray.Length;
integerArray.GetLength(0);
integerArray.GetLength(1);		

-\n: enter a new line -\t: insert a tab -\b: one char back -\\: print the char \ -\": print the char "		Special characters
int.Parse (Console.ReadLine()) double.Parse (Console.ReadLine())		Converting inputs
(int) (double)		casting

.CopyTo(array, index)
.Copy(src, dest, length)
.Clone() – we need to cast the return type.
	*Src = the source array
	*Dest = the destination array
	*length = amount of elements to copy
		Copy arrays

Array.Sort()	It sorts the elements in the array according to their type and order	Sorting an array
Clear(array,index,length)	Index – where to start Length – how many elements to clear	Array clear

Foreach (type varName in ArrName)
{
Use varName here as the current value
}
		foreach

[array name].Rank	will return the number of dimensions –in our	rank
(casting)[array name].Clone()	will make and return another copy of the matrix	clone
LIFO – last in, first out	Push – add an element Pop – remove an element	Stack
FIFO – first in last out	Enqueue – add an element Dequeue - remove an element	Queue