CME1211
Algorithms and Programming I

Strings

Assoc.Prof.Dr. Derya BIRANT

Outline

- Characters
- Strings
- String Operations
- Examples

Characters

- ***Character*** data type is used to store one alphanumeric data, such as letters, numbers, spaces, symbols, and punctuation.

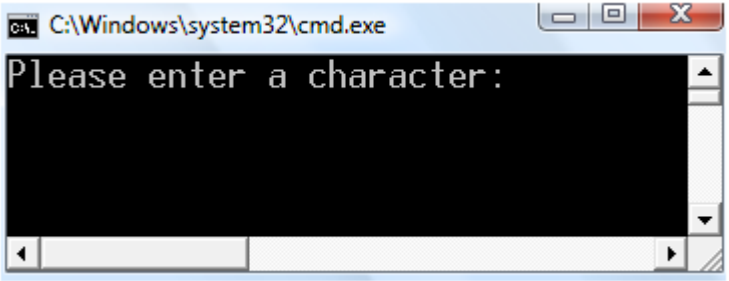
- Examples: A B a b 1 2 ; ? # % ...

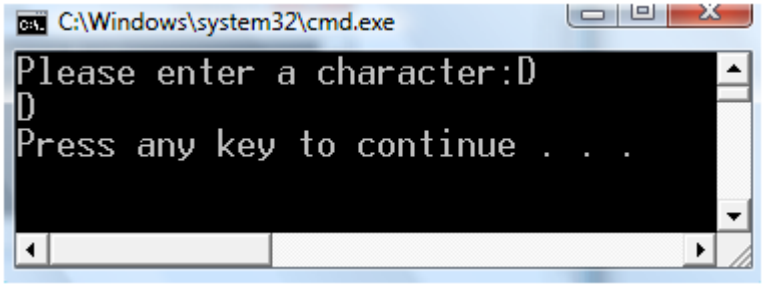
Characters in C#

- The statement **char ch;** declares a single-byte variable named "**ch**" which holds one character.

- Example:

```
char ch;
Console.Write("Please enter a character:");
ch = Convert.ToChar(Console.Read());
Console.WriteLine(ch);
```

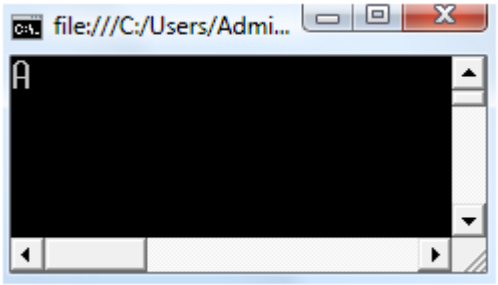




Characters in C#

- The value of a *char* is assigned in single quote (' ') using the standard assignment operator (=) .
- Example:

```
char ch;
ch = 'A';
Console.WriteLine(ch);
```

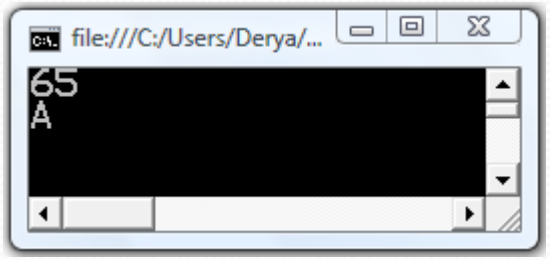


ASCII Character Codes

ASCII	Symbol	ASCII	Symbol	ASCII	Symbol	ASCII	Symbol
0	NUL	16	DLE	32	(space)	48	0
1	SOH	17	DC1	33	!	49	1
2	STX	18	DC2	34	"	50	2
3	ETX	19	DC3	35	#	51	3
4	EOT	20	DC4	36	$	52	4
5	ENQ	21	NAK	37	%	53	5
6	ACK	22	SYN	38	&	54	6
7	BEL	23	ETB	39	'	55	7
8	BS	24	CAN	40	(	56	8
9	TAB	25	EM	41	)	57	9
10	LF	26	SUB	42	*	58	:
11	VT	27	ESC	43	+	59	;
12	FF	28	FS	44	,	60	<
13	CR	29	GS	45	-	61	=
14	SO	30	RS	46	.	62	>
15	SI	31	US	47	/	63	?

ASCII	Symbol	ASCII	Symbol	ASCII	Symbol	ASCII	Symbol
64	@	80	P	96	`	112	p
65	A	81	Q	97	a	113	q
66	B	82	R	98	b	114	r
67	C	83	S	99	c	115	s
68	D	84	T	100	d	116	t
69	E	85	U	101	e	117	u
70	F	86	V	102	f	118	v
71	G	87	W	103	g	119	w
72	H	88	X	104	h	120	x
73	I	89	Y	105	i	121	y
74	J	90	Z	106	j	122	z
75	K	91	[	107	k	123	{
76	L	92	\	108	l	124	|
77	M	93	]	109	m	125	}
78	N	94	^	110	n	126	~
79	O	95	_	111	o	127	

```
Console.WriteLine(Convert.ToInt16('A'));
Console.WriteLine(Convert.ToChar(65));
```



Strings

- A ***string*** is an ordered sequence of characters.
- Examples: "Apple" "Sea"
- A ***string data type*** is a data type storing a sequence of data values, usually bytes, in which elements usually stand for characters.

Strings in C#

- A *string variable* can be declared in different ways:

```
// Declare without initializing.
string s1;

// Initialize to null.
string s2 = null;

// Initialize as an empty string.
string s3 = "";

//Initialize with an initial value.
string s4 = "Hello Word!";
```

Strings in C#

- The value of a *string is* assigned in double quotes (" ") using the standard assignment operator (=).

- Example:

```
string name;

Console.Write("Enter your name: ");
name= Console.ReadLine();
Console.WriteLine(name);

name= "Ali";
Console.WriteLine(name);
```

String Operations

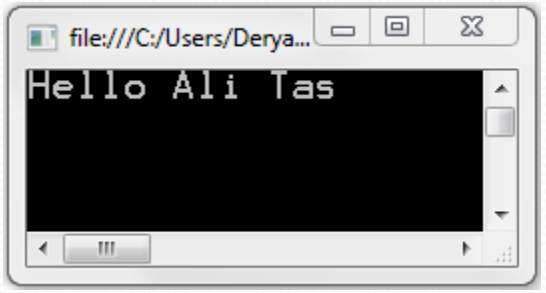
Most popular operations over strings:

- Concatenate
- Equals
- CompareTo
- Length
- Indexer
- Substring
- ToLower and ToUpper
- Split
- Indexof
- Contains
- ...

String Concatenate

String concatenation can be done using **+** operator

- Example:




```
string name = "Ali";
string surname = "Tas";
string str;
str = "Hello " + name + " " + surname;
Console.WriteLine(str);
```

Comparing Strings

There are several ways to compare strings for equality

- **Equals()**
  - Returns *true* if they are equal, otherwise *false*
- **==** and **!=** operators
  - Returns *true* or *false*
- **CompareTo()**
  - Returns 0 if equal, negative if less, positive if greater

Comparing Strings

```
string str1 = "hello";
string str2 = "mello";
```

```
if (str1.Equals(str2))
  Console.WriteLine("Equal");
else
    Console.WriteLine("Not equal");
```

```
if (str1 == str2)
  Console.WriteLine("Equal");
else
  Console.WriteLine("Not equal");
```

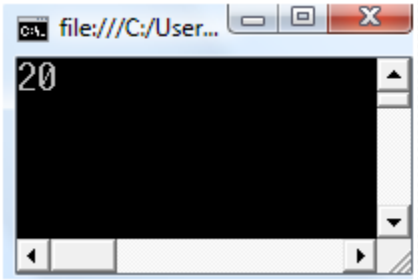
```
int result = str1.CompareTo(str2);
if (result == 0)
  Console.WriteLine("Equal");
else if (result > 0)
  Console.WriteLine("Greater");
else
  Console.WriteLine("Less");
```

String Length

- Returns the length of the string
- Useful for loops

- Example:

```
string str;
str = "Bugun hava cok guzel";
Console.WriteLine(str.Length);
```



Question

- Take two strings from the user and print the longest one.
- Example:

```
Inputs: derya
        cem
Output: derya
```

```
string s1, s2;
Console.WriteLine("enter two strings");
s1 = Console.ReadLine();
s2 = Console.ReadLine();
if (s1.Length > s2.Length)
    Console.WriteLine(s1);
else if (s2.Length > s1.Length)
    Console.WriteLine(s2);
else
    Console.WriteLine(s1 + " " + s2);
```

String Indexer

- Retrieves any character in the string using a subscript
- Example:

```
string str;
str = "Bugun hava cok guzel";

Console.WriteLine(str[6]);
Console.WriteLine(str[8]);

if (str[4]=='n')
    Console.WriteLine("YES");
```

Example

- Write a program that finds “how many times letter ‘*a*’ appears in the string”

```
string str = "Bugun hava çok güzel";
int counter = 0;

for (int i = 0; i < str.Length; i++)
   if (str[i] == 'a')
      counter++;

Console.WriteLine(counter);
```

Substring

- Retrieves a substring from this instance.
  - The substring starts at a specified character position.

    Substring(startIndex)
  - The substring starts at a specified character position and has a specified length

    Substring(startIndex, length)

- Example:

```
string str = "İstanbul soğuk, Ankara yağışlı";
string str2;

str2 = str.Substring(16, 6);
Console.WriteLine(str2);

str2 = str.Substring(16);
Console.WriteLine(str2);
```

Example

- Write a program that finds “how many times word ‘*çok*’ appears in the string”

```
string str = "Bugun hava çok ama çok çok güzel";
int counter = 0;

for (int i = 0; i < str.Length - 2; i++)
     if (str.Substring(i, 3) == "çok")
          counter++;

Console.WriteLine(counter);
```

ToLower – ToUpper

- **ToLower** : Returns a copy of the string converted to lowercase
- **ToUpper** : Returns a copy of the string converted to uppercase

- Example:

```
string str = "AlGoRitmA";
string output;

output = str.ToUpper();
Console.WriteLine(output);

output = str.ToLower();
Console.WriteLine(output);
```

Split

- Returns a string array that contains the substrings in this instance that are delimited by elements of a specified string.

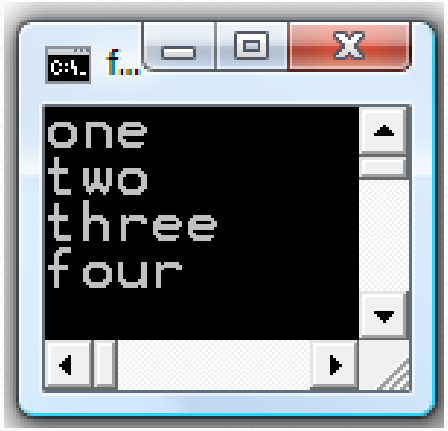
  Split(separator)

- Example:

```
string str = "one;two;three;four";
string[] words = null;

words = str.Split(';');

for (int i = 0; i < words.Length; i++)
     Console.WriteLine(words[i]);
```



Example

- Print the longest word in the string

```
string str = "I love algorithm course";
string[] words = str.Split(' ');

int max = 0;
string result = "";

for (int i = 0; i < words.Length; i++) {
    if (words[i].Length > max {
        max = words[i].Length;
        result = words[i];
    }
}
Console.WriteLine(result);
```

Char/String Search

Finds the position of a char or string

- **IndexOf** (value, startIndex)
  - ***value:*** The string to seek
  - ***startIndex:*** The search starting position (optional)

```
int index;
string letters = "abcdefabcdef";

index = letters.IndexOf('*');
Console.WriteLine(index);              // -1

index = letters.IndexOf('c');
Console.WriteLine(index);               // 2

index = letters.IndexOf("def");
Console.WriteLine(index);               // 3

index = letters.IndexOf('a', 3);
Console.WriteLine(index);                // 6
```

Contains

- Returns a boolean value indicating whether the specified string occurs within another string

  Contains(stringvalue)

- Example:

```
string str1 = "I love algorithm and programming course";
string str2 = "algorithm";

if (str1.Contains(str2))
   Console.WriteLine("yes");

if (str1.Contains("mathematics"))
   Console.WriteLine("no");
```

Example

- Write a program that finds a string is palindrome or not

ey edip adanada pide ye
anastas rulo iyi olur satsana
kalas yok kütük koy salak

```
string str = "MADAM";
bool flag = true;
for (int i = 0; i < (str.Length -1)/2; i++)
     if (str[i] != str[str.Length-i-1])
            flag = false;

if (flag)
    Console.WriteLine("Palindrome");
else
    Console.WriteLine("Not palindrome");
```

Example

- Write a program that finds the similarity percentage of two strings
- Example: "AHMET SABRİ KESGİN“
  "AHNET SAPRİ KESKİN“

18 characters, 3 characters are different

100 – ((3 / 18) * 100) % 83.3

```
string str1="AHMET SABRİ KESGİN";
string str2="AHNET SAPRİ KESKİN";
double counter = 0;

for (int i = 0; i < str1.Length; i++)
     if (str1[i] == str2[i])
         counter++;

Console.WriteLine((counter / str1.Length) * 100);
```