المحاضرة #٢: تمثيل البيانات

١ مقدمة

هنالك اصطلاحان مهمان يجب معرفتهما هما البيانات (Data) والمعلومات (Information). تمثل **البيانات** القيم أو الحقائق الأساسية وتكون عادة غير منظمة بينما **المعلومات** هي البيانات التي تم تنظيمها أو معالجتها بشكل ما لاستخدامها في حل مسألة ما. مثال على ذلك بيانات الطلاب الأساسية من الاسم والعمر والعنوان والدرجات والخ من بيانات مختلفة تخص الطالب في جامعة ما. كل البيانات السابقة يمكن تنظيمها ومعالجتها باستخدام برامج عدة للحصول على معلومات مختلفة مثل تقسيم عدد الطلاب الناجحون على أساس الفئة العمرية او التقسيم الجغرافي أو أي تقسيم أخر.

إلى وقت غير بعيد، كانت الحواسيب تتعامل بشكل خاص مع البيانات العددية والنصية فقط. لكن في هذه الأيام، تتعامل الحواسيب مع مختلف الأنواع من البيانات مثل الصوت والصور والفيديو إضافة إلى النصوص والإعداد. في النهاية، كل هذه الأنواع من البيانات يتم خزنها ومعالجتها بهيئة الإعداد الثنائية (Binary Numbers or Digits) أي الصفر والواحد.

٢ أنظمة الإعداد

يسمى نظام الإعداد الذي نستخدمه يوميا للحساب بالنظام العشري وذلك لأنه يتكون من عشرة إعداد (٠,١,٢,٣,٤,٥,٦,٧,٨,٩) ولان مرتبة كل عدد مضروبة في الرقم عشرة مرفوع للأس مرتبة ذلك العدد. فمثلا العدد ٩٤٣ يمكن تجزئته بالصورة التالية:

$$943 = 900 + 40 + 3 = 9\times10^{2} + 4\times10^{1} + 3\times10^{0}$$

بشكل مشابه يمكننا تكوين أنظمة عددية أخرى بحيث تتكون من (x) من الإعداد وتكون مرتبة كل عدد على النحو التالي:

$$d_3\times x^{2} + d_2\times x^{1} + d_1\times x^{0}$$

فمثلا النظام الثنائي (Binary System) يتكون من عددين فقط (٠و١) وكل عدد يكون مرتب حسب الأساس (٢) فمثلا العدد ١٠١:

$$101_{Two} = 1\times2^{2} + 0\times2^{1} + 1\times2^{0} = 5_{Ten}$$

وبنفس الطريق يتم تمثيل النظام الثماني (Octal System):

$$754_{Oct} = 7\times8^{2} + 5\times8^{1} + 4\times8^{0} = 492_{Ten}$$

وكذلك تمثيل نظام الأعداد السداسي عشر (Hexadecimal System) والذي يتكون من ستة عشر عدداً (٠, ١, ٢, ٣, ٤, ٥, ٦, ٧, ٨, ٩, A, B, C, D, E, F). فمثلا العدد ABC يمكن تجزئته بالصورة التالية:

$$ABC_{Hex} = A\times16^{2} + B\times16^{1} + C\times16^{0} = 2748_{Ten}$$

يمكننا أيضا القيام بالعمليات الحسابية على هذه الأنظمة كما هو الحال مع النظام العشري. فمثلا يمكننا جمع عددين بالنظام الثنائي بالشكل التالي:

$$\begin{array}{r} 101110 \\ 11011 + \\ \hline 1001001 \end{array}$$

كذلك يمكننا القيام بعملية الطرح بنفس الشكل:

$$\begin{array}{r} 111001 \\ 110 - \\ \hline 110011 \end{array}$$

في أمثلة الأنظمة السابقة رأينا كيف يتم تحويل كل الأنظمة العددية إلى النظام العشري وذلك بضرب كل عدد بأساس ذلك النظام مرفوعا بأس مرتبة ذلك العدد. ولكي نقوم بتحويل النظام العشري إلى الأنظمة الأخرى يجب عكس هذه العملية وذلك بقسمة العدد العشري على أساس ذلك النظام. فمثلا لتحويل عدد من النظام العشري (Decimal) إلى الثنائي (Binary) كالرقم ٣٧ مثلا:

37 / 2 = 18 reminder **1** LSB – Least Significant Bit
18 / 2 = 9 reminder **0**
9 / 2 = 4 reminder **1**
4 / 2 = 2 reminder **0**
2 / 2 = 1 reminder **0**
1 / 2 = 0 reminder **1** MSB – Most Significant Bit

$37_{ten} = 100101_{two}$

كذلك لتحويل العدد من العشري (Decimal) إلى السداسي عشر (Hexadecimal) نقسم على الأساس ١٦ كما في المثال التالي:

12345 / 16 = 771 reminder **9** LSB
771 / 16 = 48 reminder **3**
48 / 16 = 3 reminder **0**
3 / 16 = 0 reminder **3** MSB

$12345_{ten} = 3039_{hex}$

أما إذا أردنا تحويل الأعداد بين الأنظمة الثلاثة الثنائي والثماني والسداسي عشر فالأمر أسهل وذلك لكون أساس هذه الأنظمة هي مضاعفات العدد ٢. فمثلا لتحويل الأعداد بين النظامين الثماني (Octal) والثنائي (Binary) سيكون عبر جدول فيه كل أعداد النظام الثماني وما يعادله بالنظام الثنائي:

0	1	2	3	4	5	6	7
000	001	010	011	100	101	110	111

وعلية سيكون التحويل بين عددين من هذين النظامين بالشكل التالي:

$754_{oct} = 111101100_{two}$

حيث يتم تحويل كل عدد على حدة من الجدول السابق.

وبنفس الطريقة يتم تحويل الأعداد بين النظامين السداسي عشر (Hexadecimal) والثنائي (Binary) ولكن عبر جدول أعداد النظام السداسي عشر وما يقابله بالثنائي:

0	1	2	3	4	5	6	7
0000	0001	0010	0011	0100	0101	0110	0111
8	9	A	B	C	D	E	F
1000	1001	1010	1011	1100	1101	1110	1111

وعلية سيكون التحويل بين عددين من هذين النظامين بالشكل التالي:

$ABC_{hex} = 101010111100_{two}$

يبقى السؤال المهم. لماذا نحتاج إلى أنظمة أعداد غير النظام العشري الذي نستخدمه في كل أمور حياتنا؟ جواب هذا السؤال يكمن في ان الحواسيب لا يمكنها التعامل مع الأعداد العشرية التي

نستخدمها. لان الحواسيب عبارة عن دوائر إلكترونية رقمية ليس لديها غير حالتين لتمثيل البيانات أما إشارة فولتية عالية أو إشارة فولتية واطئة. ولذلك يتم تمثيل كل حالة بقيمة أما الـ(٠) أو (١). وهذه هي أهمية النظام الثنائي للأعداد. ولكون النظام الثنائي مكون من عددين فقط فان تمثيل الأعداد والبيانات يتطلب استخدام أعداد ثنائية كبيرة بعدد المراتب لذلك نستخدم النظامين الثماني والسداسي عشر لغرض جعل الأعداد أصغر حجما.

شكل ١: الإشارة الرقمية

مصدر الشكل كتاب *Computer Science Illuminated* by Nell B. Dale

تسمى أصغر وحدة لخزن البيانات الرقمية بـ البت (**Bit**) وهي أما صفر (٠) أو واحد (١). كذلك تدعى كل ٨ بت بـ البايت (**Byte**).

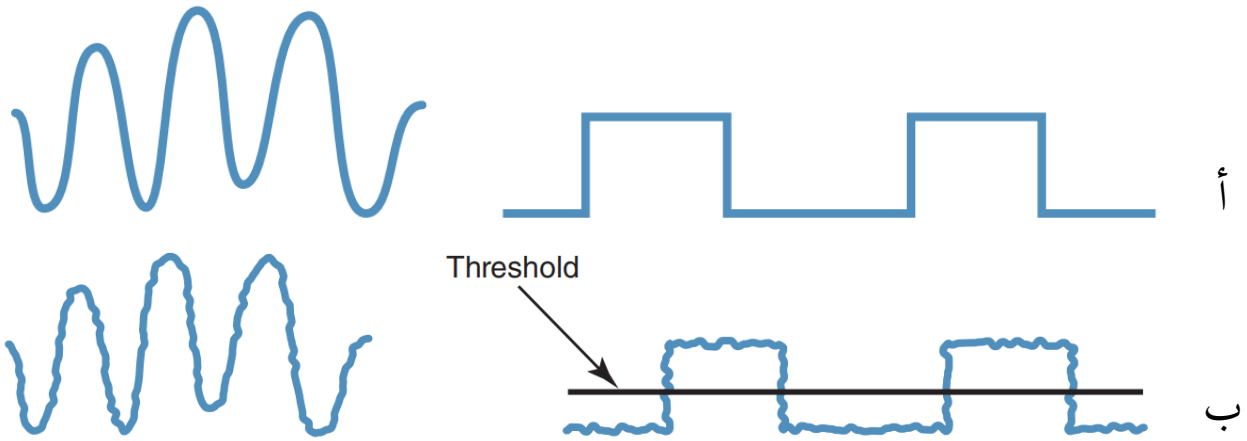
٣ بيانات الحواسيب

٣-١ البيانات التناظرية والرقمية

يمكن تمثيل البيانات بواحدة من طريقتين: تناظرية (Analog) او رقمية (Digital). **البيانات التناظرية** هي تمثيل متواصل للبيانات أي كما هو تمثيل البيانات في الواقع الحياتي. بينما **البيانات الرقمية** هي تمثيل متقطع وذلك بتقطيع المعلومات إلى عناصر منفصلة.

ان المحرار الزئبقي هو جهاز تناظري حيث يتدفق الزئبق داخل المحرار باتجاه يتناسب مع درجة الحرارة وبشكل مستمر ومتواصل. أن البيانات التناظرية متناسبة مع البيانات في العالم من حولنا من حيث كونها مستمرة وغير محدودة. بما أن الحواسيب لا يمكنها التعامل مع البيانات التناظرية لذلك نقوم برقمنة (digitize) البيانات وذلك بتجزئتها إلى قطع وتمثيل تلك القطع بشكل منفصل. حيث تمثل هذه العناصر المنفصلة بواسطة الأرقام الثنائية (الصفر والواحد). يستخدم النظام الثنائي لتمثيل البيانات الثنائية إلكترونيا بسبب سهولة تمثيلها حيث يتطلب ذلك تميّز حالتين فقط للبيانات. بينما من الصعب للإشارات الإلكترونية إن تميّز البيانات التناظرية حيث أن أي تذبذب بسيط في الإشارات الإلكترونية سيتسبب في فقدان معظم البيانات. على العكس من ذلك فان الإشارات الرقمية تمثّل بحالتين متباعدتين تسمى تضمين شفرة النبضة (Pulse-Code

Modulation - PCM). يمكن للإشارة الرقمية أن تضعف أيضا لكن يجب ان يكون مقدار الضعف كبير يتجاوز قيمة معينة تسمى العتبة (Threshold) بحيث لا يمكن تميّز حالة الصفر من الواحد.



شكل ٢: الإشارة الرقمية (يمين) والتناظرية (يسار). (أ) بدون ضوضاء (ب) مشوهتان بفعل الضوضاء

مصدر الشكل كتاب *Computer Science Illuminated* by Nell B. Dale

٢.٣ التمثيل الثنائي للبيانات

كما هو مذكور أعلاه فان النظام الثنائي لدية قيمتان فقط أما الصفر (٠) او الواحد (١). تسمى إحدى هذه القيم بـ البت (bit) وهي أصغر وحدة للبيانات الرقمية. لكي نمثل أكثر من شيئين نحتاج أكثر من بت واحد. مثلا فان ٢ بت قادر على تمثيل ٤ أشياء بالقيم التالية: ٠٠, ٠١, ١٠, ١١. وبشكل عام لتمثيل 2^n من الأشياء فان ذلك يتطلب n عدد من البت.

٣.٣ تمثيل البيانات العددية

تعتبر البيانات العددية (الأرقام الصحيحة والحقيقية) النوع الأكثر استخداما في نظام الحاسوب. على العكس من أنواع البيانات الأخرى فان تمثيل البيانات العددية لا يتطلب طريقة تحويل معقدة ولذلك لان الأعداد الثنائية هي بيانات عدديه أيضاً.

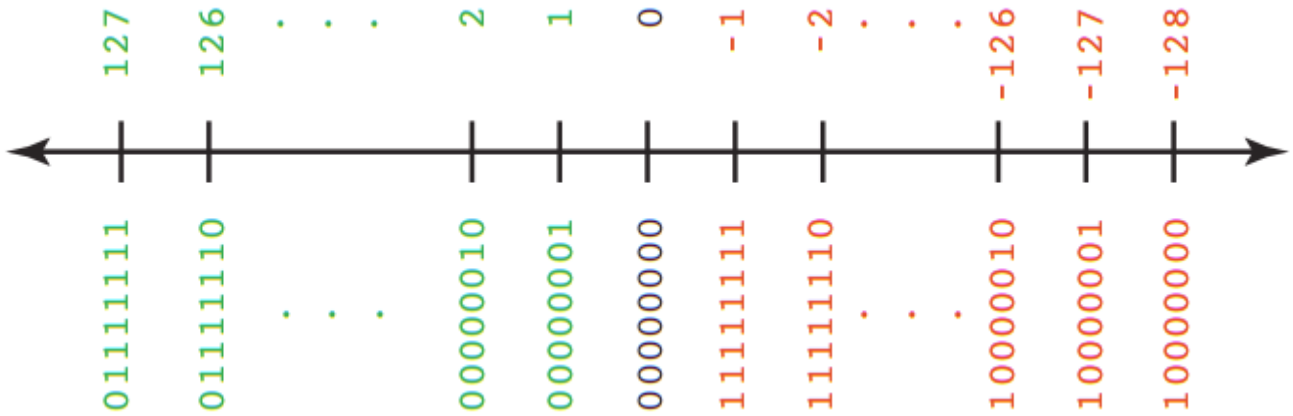
١.٣.٣ تمثيل الأعداد الصحيحة

هنالك طريقتين لتمثيل الأعداد الصحيحة بناء على نوع هذه الأعداد: الأعداد الموجبة فقط والأعداد الموجبة والسالبة.

في حالة افتراض أن جميع الأعداد موجبة فان تمثيل الأعداد يكون مباشر وبدون أي تعقيد. مثلا لو افترضنا أن نمثل كل عدد بـ٨ بت من الأعداد الثنائية:

5 + 3 = 8
5 = 00000101 ← 8 bit
3 = 00000011 +
~~00001000~~ = 8

أما في حالة الأعداد الموجبة والسالبة فيجب علينا تمثيل الإشارة إضافة إلى العدد. فلو افترضنا أن تمثيل العدد يتكون من ٨ بت كما في مثال أعلاه فسيكون لدينا ٧ بت لتمثيل الأعداد وواحد بت للإشارة. يسمى البت المسؤول عن إشارة العدد ببت الإشارة (Sign bit) ويكون موقعة في أقصى يسار العدد. في الشكل ٢ أدناه توضيح كيفية تمثيل الأعداد الموجبة والسالبة بـ٨ بت من الأعداد الثنائية. لاحظ كيف أن قيمه بت الإشارة الأخير (على اليسار) يساوي صفر (٠) لكل الأعداد الموجبة وواحد (١) لكل الأعداد السالبة.



شكل ٣: الأعداد الموجبة والسالبة ممثلة بـ٨ بت من الأعداد الثنائية

مصدر الشكل كتاب *Computer Science Illuminated* by Nell B. Dale

تسمى طريقة تمثيل الأعداد السابقة بالمتمم الثنائي (Two's Complement). وهي الطريقة التي يمكننا بها معرفة العدد السالب المقابل للموجب وذلك بأخذ بقلب (Invert) العدد ثم اضافة له العدد واحد كما في المثال التالي:

+2 = 00000010 → invert → 11111101
00000001 +
~~11111110~~ = -2

يمكننا إيجاد العدد السالب عن طريف تطبيق المعادلة التالية:

العدد السالب = 2^n – العدد الموجب

فمثلا لو كان لدينا العدد بالنظام الثنائي ١٠١٠ فأن كان تمثيله مطلوب بالأعداد الموجبة فقط (Unsigned Numbers) فان إيجاد العدد بالنظام العشري سيكون مباشرة بنفس الطريقة المذكورة في موضوع تحويل الأعداد بين الأنظمة:

$$1010 = 1\times2^3 + 0\times2^2 + 1\times2^1 + 0\times2^0 = 8+0+1+0 = 9$$

أما إذا كان تمثيله بالأعداد الموجبة والسالبة (Signed Numbers) فان العدد سالب لكون العدد في أقصى اليسار هو واحد (١٠١٠). فيمكننا إيجاد العدد السالب أما بطريقة المتمم الثنائي (Two's Complement):

$$\mathbf{1}010 \rightarrow \text{invert} \rightarrow 0101 + 0001 = 0110 = 6 \rightarrow -6$$

او يمكننا إيجاد العدد السالب عن طريق تطبيق المعادلة: العدد السالب = 2^n – العدد الموجب، حيث n تمثل عدد الأرقام الكلية لتمثيل العدد وهي في حالة مثالنا هنا أربعة أرقام وبالتالي فان المعادلة تصبح العدد السالب = ١٦- العدد الموجب:

$$1010_{two} = 10_{ten} \rightarrow 16 - 10 = 6 \rightarrow -6$$

٢-٣-٣ تمثيل الإعداد الحقيقية

يعّرف الرقم الحقيقي بالقيمة التي قد تحتوي على جزء عُشري (كسور). أي إن العدد الحقيقي يتكون من جزء صحيح وجزء عُشري. وقد يكون أحدهما صفر. مثل الأعداد في نظام الأعداد العشرية: ١٠٤.٣٢, ٠.٠٠٩٩٩٩, ٣٥٧٠٠, و ٣.١٤١٥٩.

لمعرفة كيفية تمثيل الأعداد الثنائية (Binary Numbers) في أجهزة الحواسيب يجب ملاحظة الصيغة العلمية لتمثيل الأعداد العشرية:

الإشارة × العدد × $10^{\text{الأس}}$

أمثلة على تمثيل الإعداد بالصيغة العلمية:

العدد الحقيقي	الصيغة العلمية
12001.00	1.2001×10^{4}
-120.01	-1.2001×10^{2}
0.12000	1.2×10^{-1}
-123.10	-1.231×10^{2}
155555000.00	1.5555×10^{8}

وبشكل مشابه فان تمثيل الأعداد الثنائية تأخذ صيغة مشابهة:

الاشارة × العدد × $2^{\text{الأس}}$

نلاحظ استبدال الأساس ١٠ بالأساس ٢ للأعداد الثنائية. امثلة على تمثيل الاعداد الثنائية بالصيغة العلمية:

العدد الحقيقي	الصيغة العلمية
11001.00	1.1001×2^{4}
-110.01	-1.1001×2^{2}
0.11000	1.1×2^{-1}
-111.10	-1.111×2^{2}
1111111000.00	1.11111×2^{8}

٣-٣-٣ الفيض (Overflow)

يحدث الفيض عندما يكون الناتج أكبر من عدد البت المعينة للنتيجة فمثلا إذا كان لدينا ٤ بت لتمثيل الأعداد العشرية ١٥ و١:

$$1111 + 0001 = 10000$$

نلاحظ بان الناتج خمسة أرقام، ولكن كون عدد البت المحددة لتمثيل الأعداد الثنائية هي ٤ بت فان الناتج سيكون صفر بدلا من الناتج أعلاه لعدم كفاية عدد البت لتمثيل الناتج. هذه الحالة تسمى بالفيض (Overflow).

٣-٤ تمثيل النصوص

إن أي وثيقة نصية تتكون من فقرات وجمل وكلمات وأحيانا رموز منفردة. لتمثيل أي نص لابد لنا تمثيل كل الأحرف والأرقام وأي رمز متحمل الاستخدام بما يقابلها من أعداد ثنائية. يطلق على المجموعة السابقة الذكر من الرموز وما يمثلها من أرقام ثنائية بمجموعة الترميز (Character Set). استخدمت العديد من مجموعات الترميز خلال السنين السابقة، ولكن القليل منها بقي يستخدم لحد الآن. من مجموعات الترميز الشائعة هي ASCII وUnicode.

٣-٤-١ مجموعة الترميز ASCII

تستخدم مجموعة الترميز (ASCII - American Standard Code for Information Interchange) ٧ بت لتمثيل الأحرف والرموز مما يتيح لها تمثيل ١٢٨ رمزا. أما البت الثامن فقد كان يساعد للتأكد من سلامة نقل البيانات. لاحقا تطور ASCII ليستخدم ٨ بت لتمثيل الرمز الواحد مما يتيح لها تمثيل ٢٥٦ رمزا.

٣-٤-٢ مجموعة الترميز Unicode

أن الإصدار الموسع من ASCII ذي ٢٥٦ رمزا قادر على تمثيل أحرف ورموز اللغة الإنكليزية، ولكنة ليس كافيا لبقية اللغات العالمية. وهذا هو الهدف من تكون مجموعة الترميز Unicode لتمثيل كل الرموز في كل لغات العالم حتى اللغات الصينية. بشكل عام يستخدم الترميز Unicode 16 بت لكل رمز مما يسمح لتمثيل عدد أكبر من الرموز من ASCII.

ASCII value	Char-acter	ASCII value	Char-acter	ASCII value	Char-acter	ASCII value	Char-acter	ASCII value	Char-acter	ASCII value	Char-acter
32	space	48	0	64	@	80	P	96	`	112	p
33	!	49	1	65	A	81	Q	97	a	113	q
34	"	50	2	66	B	82	R	98	b	114	r
35	#	51	3	67	C	83	S	99	c	115	s
36	$	52	4	68	D	84	T	100	d	116	t
37	%	53	5	69	E	85	U	101	e	117	u
38	&	54	6	70	F	86	V	102	f	118	v
39	'	55	7	71	G	87	W	103	g	119	w
40	(	56	8	72	H	88	X	104	h	120	x
41	)	57	9	73	I	89	Y	105	i	121	y
42	*	58	:	74	J	90	Z	106	j	122	z
43	+	59	;	75	K	91	[	107	k	123	{
44	,	60	<	76	L	92	\	108	l	124	|
45	-	61	=	77	M	93	]	109	m	125	}
46	.	62	>	78	N	94	^	110	n	126	~
47	/	63	?	79	O	95	_	111	o	127	DEL

شكل ٤: جزء من مجموعة الترميز ASCII

مصدر الشكل كتاب *Computer Organization and Design, RISC-V Edition* by Patterson and Hennessy

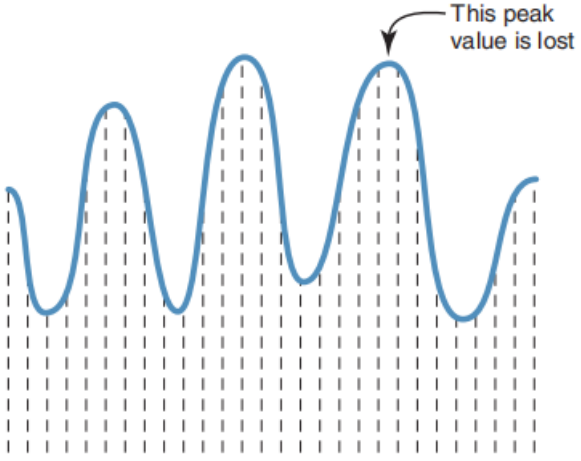
Code (Hex)	Character	Source
0041	A	English (Latin)
042F	Я	Russian (Cyrillic)
0E09	ก	Thai
13EA	Ꮺ	Cherokee
211E	℞	Letterlike symbols
21CC	⇌	Arrows
282F	⠯	Braille
345F	㑟	Chinese/Japanese/ Korean (common)

شكل٥: جزء من مجموعة الترميز Unicode لعدة لغات عالمية

مصدر الشكل كتاب *Computer Science Illuminated* by Nell B. Dale

٣-٥ تمثيل البيانات الصوتية

إن الموجات الصوتية عبارة عن إشارات تناظرية مستمرة. لتمثيل هذه الإشارات في أجهزة الحاسوب يجب تقطيع هذه الإشارات إلى أجزاء يمكن التعامل معها. الطريقة التي تقوم بذلك هي رقمنه الإشارات الصوتية التناظرية وذلك بقراءة قيم محددة لتلك الإشارات لتكوين متسلسلة من القيم العددية المنفصلة (discrete). تسمى هذه الطريقة بالنمذجة أو أخذ العينات (Sampling). بشكل عام يعتبر معدل النمذجة بمقدار ٤٠٠٠٠ نموذج لكل ثانية كافياً لتكوين صوت مفهوم بشريا. إذا كان معدل النمذجة اقل من ذلك فان الأذن البشرية تبدأ بسماع تشويش. أما إذا كان معدل النمذجة أعلى من ذلك فان جودة الصوت ستكون جيدة. وإذا زاد معدل النمذجة عن حد معين فلن يكون له أهمية لان الأذن البشرية لن تستطيع تمييز الفرق.



شكل٦: عملية النمذجة (Sampling) الخطوط المتقطعة تمثل النقاط التي تأخذ من الإشارة الأصلية المستمرة

مصدر الشكل كتاب *Computer Science Illuminated* by Nell B. Dale

تحفظ الملفات الصوتية في الحاسوب بعدة صيغ تختلف فيما بينها من حيث اختلاف طريقة ضغط البيانات. أشهر هذه الصيغ: WAV, AU, AIFF, VQF, and MP3. تعتبر الصيغة MP3 الأكثر استخداما في الوقت الحالي. ويعود السبب الى قدرة هذه الصيغة على ضغط ملفات الصوت إلى أحجام صغيرة أكثر من الصيغ الأخرى. تمثل الـ MP3 مختصر إلى MPEG-2 الطبقة الثالثة. أما (MPEG) فتشير إلى (Moving Picture Experts Group) وهي لجنة دولية تطور معايير لضغط ملفات الصوت والفيديو الرقميين.

لتحويل الملفات الصوتية الأخرى إلى ملف MP3 تمر بثلاثة مراحل: الأولى تحليل الترددات ومقارنتها بالنماذج الرياضية لألية استماع البشر للصوت. ثم يتم إهمال المعلومات التي لا يمكن للأذن البشرية سماعها. أخيراً يتم ضغط الملف بواسطة خوارزمية تشفير كفؤه.

شكل٧: مراحل تحويل الملف الصوتي إلى الصيغة MP3

٦.٣ تمثيل الصور والرسوم

يتم تمثيل الصور في الحاسوب على شكل نقاط (dots) تدعى كل نقطة بكسل (Pixel). كل بكسل (Pixel) تمثل لون واحد. عدد الـ بكسل المستخدمة لتمثيل صورة تسمى الدقة (Resolution). تمثل جميع الألوان باستخدام ثلاث ألوان: الأحمر والأخضر والأزرق. يتدرج كل لون من ٠ إلى ٢٥٥ درجة مما يمّكننا من تمثيل ١٦٠٧ مليون لون مختلف وهي أكثر من قدرة العين البشرية على تمييز الألوان. المدى من ٠ إلى ٢٥٥ يحتاج هذا المدى من الألوان إلى ٨ بت ويطلق علية عمق اللون (Color Depth).

من أكثر صيغ ملفات الصور شيوعا هي BMP, GIF, JPEG, and PNG. تدعم الصيغة BMP تمثيل الألوان بعمق ٢٤-بت (٨ لكل لون). بينما لا تستطيع GIF التمييز لأكثر من ٢٥٦ لون فقط. الصيغة JPEG هي الأكثر شيوعا وذلك لأنها تحاكي العين البشرية لرؤية الصور ولأنها تضغط الصور لتكون أصغر حجما عن خزن ملفاتها. أما الصيغة PNG فهي تهدف لاستبدال GIF وذلك بتوفير عمق أكثر للألوان.

تستخدم تقنية الرسوم الاتجاهية (Vector graphics) لتمثيل الرسوم بدلا من الـ بيكسل. تمثّل الرسوم بهذه الصيغة بواسطة الخطوط والإشكال الهندسة. يحتوي ملف الرسوم الاتجاهية على معلومات عن اتجاه الخط وسمكة ولونه. من صيغ ملفات الرسوم أشهر صيغتين هما Adobe Flash Scalable Vector Graphics (SVG)

٣-٧ تمثيل الفيديو

أن مقاطع الفيديو هي عبارة عن صور تظهر بشكل متسلسل زمنيا. أن جودة الفيديو تعتمد على طريقة تقنيات ضغط الفيديو المستخدمة والتي تدعى بـ (VIDEO CODECS). إن كلمة CODEC أتت من الكلمتين COmpressor/DECompressor. هنالك نوعان من طرق ضغط الفيديو: الضغط الزماني (Temporal compression) والضغط المكاني (Spatial compression). يعتمد الضغط الزماني على تتبع التغيرات بين الصور المتلاحقة (frames) وبالتالي إهمال التفاصيل المتكررة بينهم. بينما يعتمد الضغط المكاني على حذف التفاصيل المكررة داخل الصورة (Frame) الواحدة بطريقة مشابهه نوعا ما لتقنيات ضغط الصور الثابتة حيث يعمل على تجميع الـ بيكسل التي لها نفس اللون في مربعات. تعتبر تقنيات ضغط التالية الطرق الأكثر شيوعا: Sorenson, Cinepak, MPEG, and Real Video.

في هذه المحاضرة تناولنا طرق تمثيل البيانات المختلفة: الأعداد والرموز والصوت والصور والفيديو إلى أعداد ثنائية (Binary Numbers) وذلك لان الحاسوب يمكنه تمثيل حالتين فقط اما صفر او واحد. في المحاضرة القادمة سنستعرض مكونات الحاسوب المادية (Hardware) من حيث تفسير معنى مواصفات الحواسيب الى الأسس النظرية المستندة عليها.