Activity 6

Battleships—*Searching Algorithms*

Summary

Computers are often required to find information in large collections of data. They need to develop quick and efficient ways of doing this. This activity demonstrates three different search methods: linear searching, binary searching and hashing.

Curriculum Links

- ✓ Mathematics: Number Level 3 and up. Exploring numbers: Greater than, less than and equal to
- ✓ Geometry Level 3 and up. Exploring shape and space: Co-ordinates

Skills

- ✓ Logical reasoning

Ages

- ✓ 9 years and up

Materials

Each child will need:

- ✓ Copy of battleships games
  - 1A, 1B for game 1
  - 2A, 2B for game 2
  - 3A, 3B for game 3

- ✓ You may also need a few copies of the supplementary game sheets, 1A', 1B', 2A', 2B', 3A', 3B'.

© 2002 Computer Science Unplugged (www.unplugged.canterbury.ac.nz)

Battleships

Introductory Activity

1. Choose about 15 children to line up at the front of the classroom. Give each child a card with a number on it (in random order). Keep the numbers hidden from the rest of the class.

2. Give another child a container with four or five sweets in it. Their job is to find a given number. They can "pay" to look at a particular card. If they find the correct number before using all their sweets, they get to keep the rest.

3. Repeat if you wish to.

4. Now shuffle the cards and give them out again. This time, have the children sort themselves into ascending order. The searching process is repeated.

   If the numbers are sorted, a sensible strategy is to use just one "payment" to eliminate half the children by having the middle child reveal their card. By repeating this process they should be able to find the number using only three sweets. The increased efficiency will be obvious.

Activity

The children can get a feel for how a computer searches by playing the battleship game. As they play the game, get them to think about the strategies they are using to locate the ships.

Photocopiable for classroom use only.
© 2005 Computer Science Unplugged (www.unplugged.canterbury.ac.nz)

Battleships—A *Linear* Searching Game

Read the following instructions to the children

1. Organise yourselves into pairs. One of you has sheet 1A, the other sheet 1B. Don't show your sheet to your partner!

2. Both of you circle one battleship on the top line of your game sheet and tell your partner its number.

3. Now take turns to guess where your partner's ship is. (You say the letter name of a ship and your partner tells you the number of the ship at that letter.)

4. How many shots does it take to locate your partner's ship? This is your score for the game.

   (Sheets 1A' and 1B' are extras provided for children who would like to play more games or who "inadvertently" see their partner's sheet. Sheets 2A', 2B' and 3A', 3B' are for the later games.)

Follow Up Discussion

1. What were the scores?

2. What would be the minimum and maximum scores possible? (They are 1 and 26 respectively, assuming that the children don't shoot at the same ship twice. This method is called 'linear search', because it involves going through all the positions, one by one.)

© 2002 Computer Science Unplugged (www.unplugged.canterbury.ac.nz)

Battleships—A *Binary* Searching Game

Instructions

The instructions for this version of the game are the same as for the previous game but the numbers on the ships are now in ascending order. Explain this to the children before they start.

1. Organise yourselves into pairs. One of you has sheet 2A, the other sheet 2B. **Don't** show your sheet to your partner!

2. Both of you circle one battleship on the top line of your game sheet and tell your partner its number.

3. Now take turns to guess where your partner's ship is. (You say the letter name of a ship and your partner tells you the number of the ship at that letter.)

4. How many shots does it take to locate your partner's ship? This is your score for the game.

Follow Up Discussion

1. What were the scores?

2. What strategy did the low scorers use?

3. Which ship should you choose first? (The one in the middle tells you which half of the line the chosen ship must be in.) Which location would you choose next? (Again the best strategy is always to choose the middle ship of the section that must have the selected ship.)

4. If this strategy is applied how many shots will it take to find a ship? (Five at most).

   This method is called 'binary search', because it divides the problem into two parts.

Photocopiable for classroom use only.
© 2005 Computer Science Unplugged (www.unplugged.canterbury.ac.nz)

Battleships—A Search Game using *Hashing*

Instructions

1. Each take a sheet as in the previous games and tell your partner the number of your chosen ship.

2. In this game you can find out which column (0 to 9) the ship is in. You simply add together the digits of the ship's number. The last digit of the sum is the column the ship is in. For example, to locate a ship numbered 2345, add the digits 2+3+4+5, giving 14. The last digit of the sum is 4, so that ship must be in column 4. Once you know the column you need to guess which of the ships in that column is the desired one. This technique is called 'hashing', because the digits are being squashed up ("hashed") together.

3. Now play the game using this new searching strategy. You may like to play more than one game using the same sheet—just choose from different columns.

   (Note that, unlike the other games, the spare sheets 3A' and 3B' must be used as a pair, because the pattern of ships in columns must correspond.)

Follow Up Discussion

1. Collect and discuss scores as before.

2. Which ships are very quick to find? (The ones that are alone in their column.) Which ships may be harder to find? (The ones whose columns contain lots of other ships.)

3. Which of the three searching processes is fastest? Why?

   What are the advantages of each of the three different ways of searching? (The second strategy is faster than the first, but the first one doesn't require the ships to be sorted into order. The third strategy is usually faster than the other two, but it is possible, by chance, for it to be very slow. In the worst case, if all the ships end up in the same column, it is just as slow as the first strategy.)

© 2002 Computer Science Unplugged (www.unplugged.canterbury.ac.nz)

Extension Activities

1. Have the children make up their own games using the three formats. For the second game they must put the numbers in ascending order. Ask how they might make the Hashing Game very hard. (The hardest game is when all the ships are in the same column.) How could you make it as easy as possible? (You should try to get the same number of ships into each column.)

2. What would happen if the ship being sought wasn't there? (In the Linear Search game it would take 26 shots to show this. In the Binary Search game you would need five shots to prove this. When using the Hash System it would depend on how many ships appeared in the relevant column.)

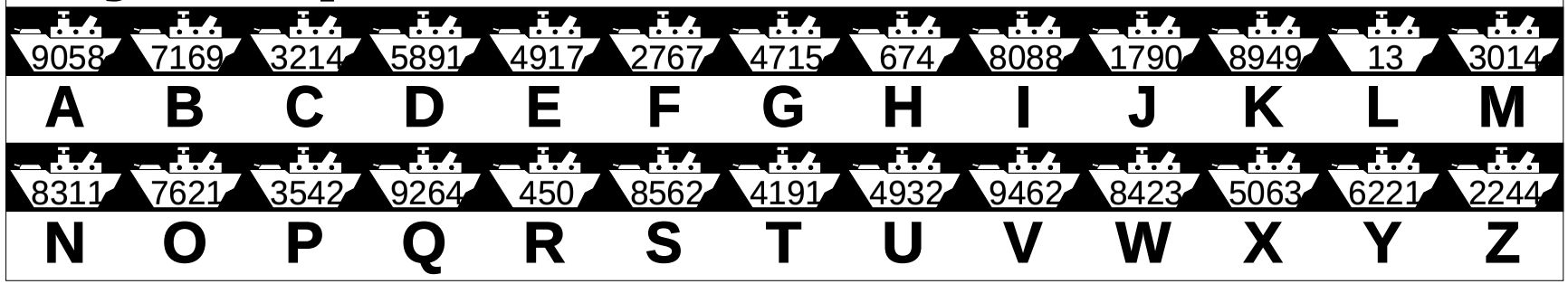
3. Using the Binary Search strategy how many shots would be required if there were a hundred locations (about six shots), a thousand locations (about nine), or a million (about nineteen)? (Notice that the number of shots increases very slowly compared to the number of ships. One extra shot is required each time the size doubles, so it is proportional to the logarithm of the number of ships.)

Photocopiable for classroom use only.
© 2005 Computer Science Unplugged (www.unplugged.canterbury.ac.nz)

Photocopiable for classroom use only.
© 2002 Computer Science Unplugged (www.unplugged.canterbury.ac.nz)

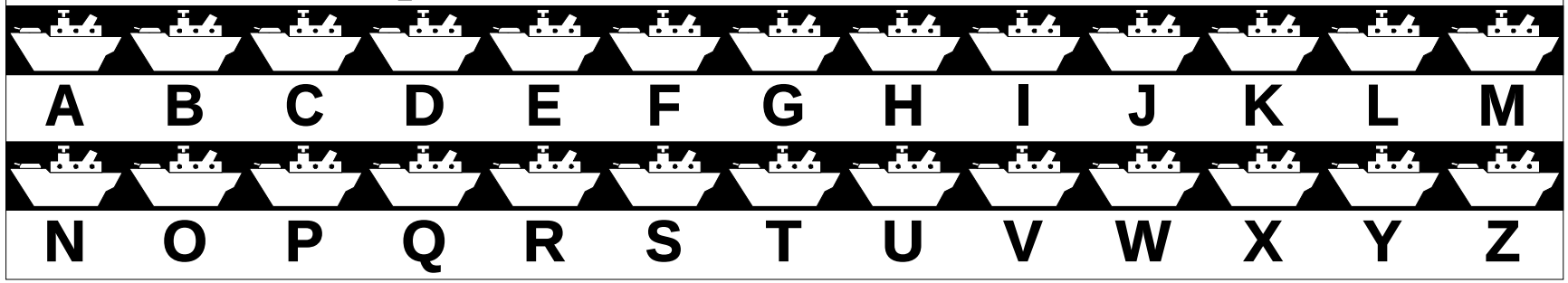
My Ships

Number of Shots Used:



Your Ships

Number of Shots Used:



1A

1B

My Ships

Number of Shots Used:

1630	9263	4127	405	4429	7113	3176	4015	7976	88	3465	1571	8625
A	B	C	D	E	F	G	H	I	J	K	L	M
2587	7187	5258	8020	1919	141	4414	3056	9118	717	7021	3076	3336
N	O	P	Q	R	S	T	U	V	W	X	Y	Z

Your Ships

Number of Shots Used:

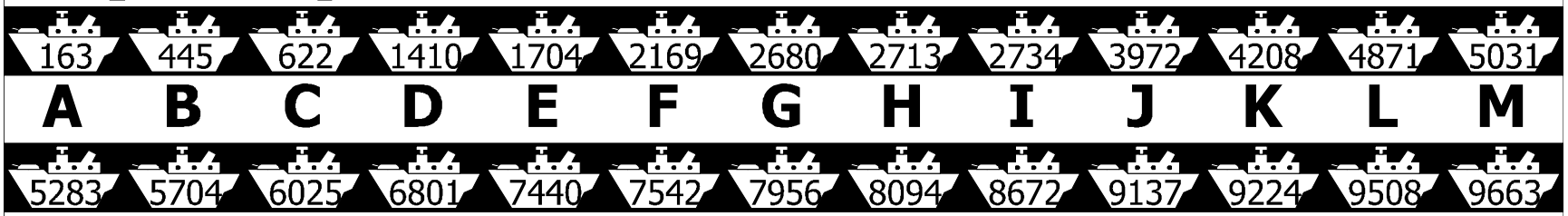
A	B	C	D	E	F	G	H	I	J	K	L	M
N	O	P	Q	R	S	T	U	V	W	X	Y	Z

Photocopiable for classroom use only.
© 2005 Computer Science Unplugged (www.unplugged.canterbury.ac.nz)

Photocopiable for classroom use only.
© 2002 Computer Science Unplugged (www.unplugged.canterbury.ac.nz)

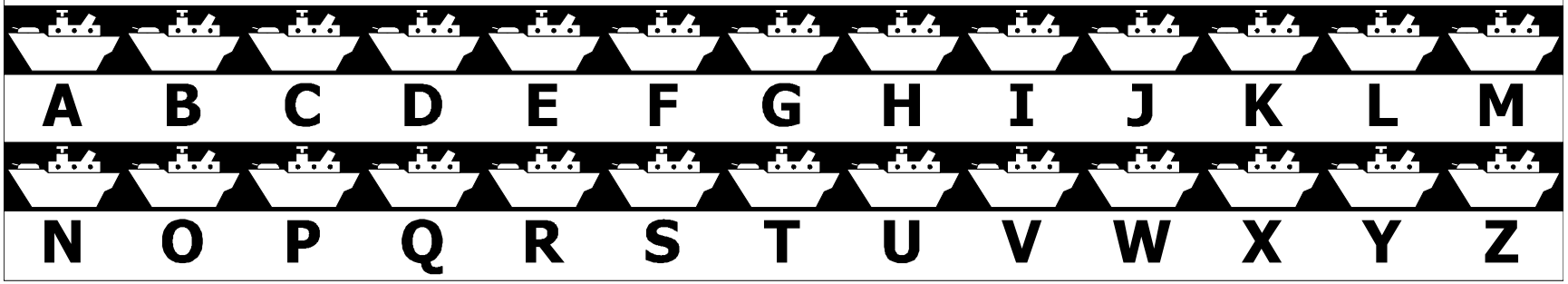
My Ships

Number of Shots Used:



Your Ships

Number of Shots Used:



2A

2B

My Ships

Number of Shots Used:

33	183	730	911	1927	1943	2200	2215	3451	3519	4055	5548	5655
A	B	C	D	E	F	G	H	I	J	K	L	M
5785	5897	5905	6118	6296	6625	6771	6831	7151	7806	8077	9024	9328
N	O	P	Q	R	S	T	U	V	W	X	Y	Z

Your Ships

Number of Shots Used:

A	B	C	D	E	F	G	H	I	J	K	L	M
N	O	P	Q	R	S	T	U	V	W	X	Y	Z

Photocopiable for classroom use only.
© 2005 Computer Science Unplugged (www.unplugged.canterbury.ac.nz)

Photocopiable for classroom use only.
© 2002 Computer Science Unplugged (www.unplugged.canterbury.ac.nz)

My Ships

Number of Shots Used:

0	1	2	3	4	5	6	7	8	9
A 9047	C 3080		E 5125	H 8051	L 7116	O 6000	R 9891		W 1062
B 1829	D 9994		F 1480	I 1481	M 8944	P 7432	S 1989	V 4392	X 2106
			G 8212	J 4712	N 4128	Q 4110	T 2050		Y 5842
				K 6422			U 8199		Z 7057

Your Ships

Number of Shots Used:

0	1	2	3	4	5	6	7	8	9
A	E	H	K	L		O	R	V	Y
B	F	I		M		P	S	W	Z
C	G	J		N		Q	T	X	
D							U		

3A

© 2005 Computer Science Unplugged (www.unplugged.canterbury.ac.nz)

Photocopiable for classroom use only.

My Ships

Number of Shots Used:

0	1	2	3	4	5	6	7	8	9
A 9308	E 6519	H 1524	K 4135	L 9050		O 4200	R 3121	V 2385	Y 1990
B 1478	F 2469	I 8112		M 1265		P 7153	S 9503	W 5832	Z 2502
C 8417	G 5105	J 2000		N 5711		Q 6028	T 1114	X 1917	
D 9434							U 7019		

Your Ships

Number of Shots Used:

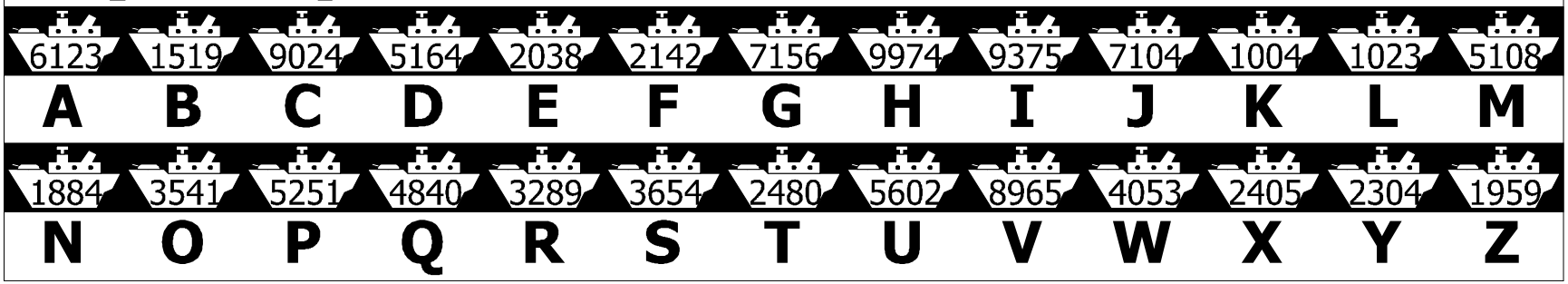
0	1	2	3	4	5	6	7	8	9
A	C		E	H	L	O	R	V	W
B	D		F	I	M	P	S		X
			G	J	N	Q	T		Y
				K			U		Z

3B

Photocopiable for classroom use only.
© 2002 Computer Science Unplugged (www.unplugged.canterbury.ac.nz)

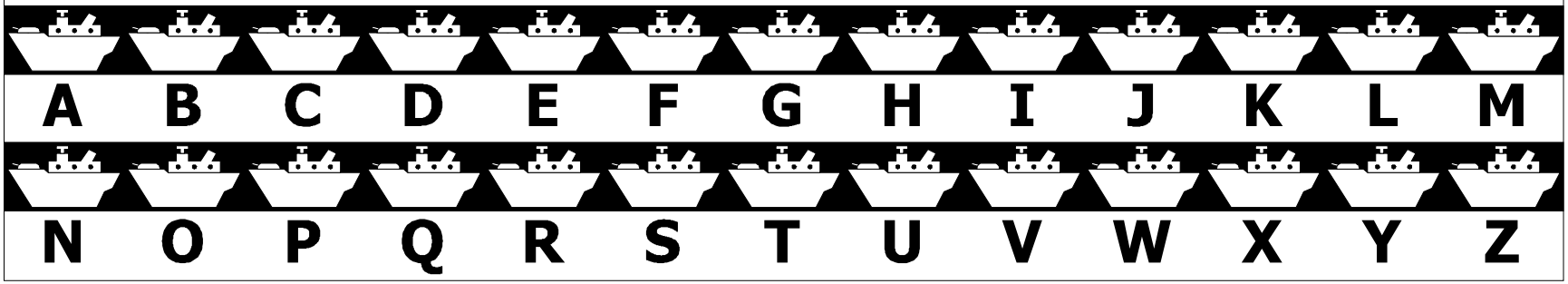
My Ships

Number of Shots Used:



Your Ships

Number of Shots Used:



1A′

1B′

My Ships

Number of Shots Used:

2387	9003	3951	5695	1284	4761	7118	1196	1741	3791	3405	3132	6682
A	B	C	D	E	F	G	H	I	J	K	L	M
9493	9864	7359	1250	7036	2916	7562	9299	8910	6713	5173	8617	4222
N	O	P	Q	R	S	T	U	V	W	X	Y	Z

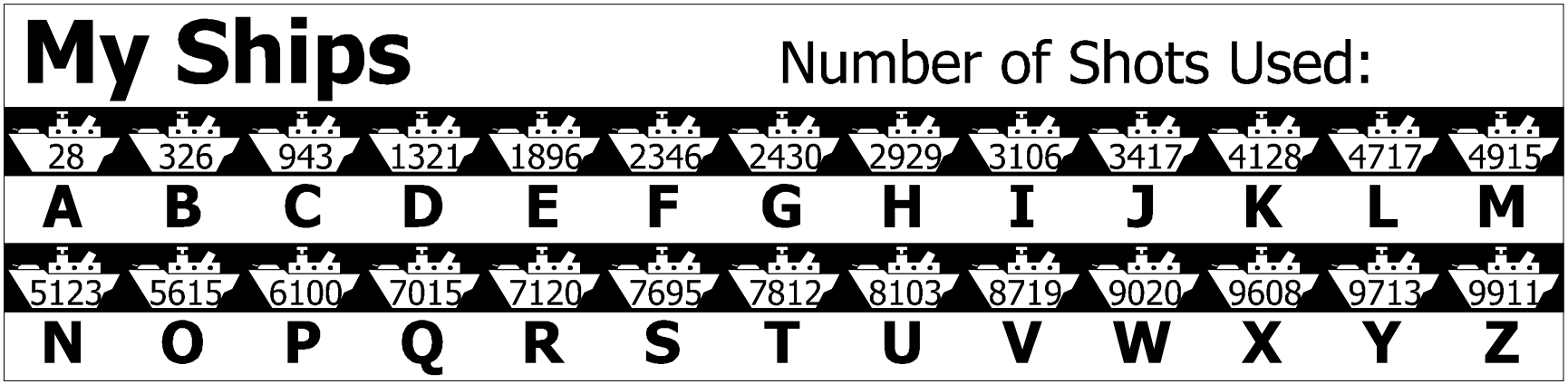
Your Ships

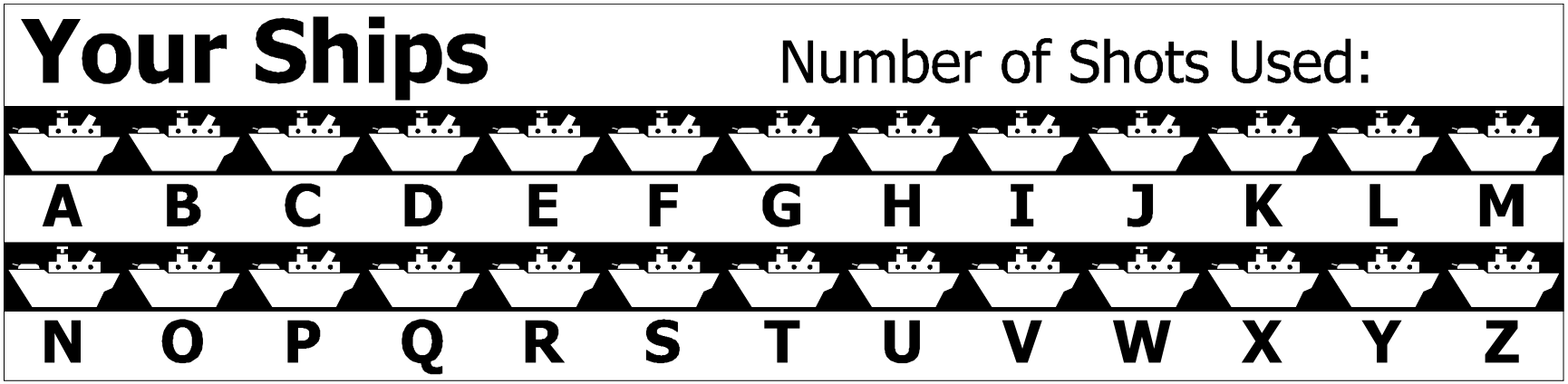
Number of Shots Used:

A	B	C	D	E	F	G	H	I	J	K	L	M
N	O	P	Q	R	S	T	U	V	W	X	Y	Z

Photocopiable for classroom use only.
© 2005 Computer Science Unplugged (www.unplugged.canterbury.ac.nz)

Photocopiable for classroom use only.
© 2002 Computer Science Unplugged (www.unplugged.canterbury.ac.nz)





2A′

2B′

My Ships

Number of Shots Used:

56	194	306	1024	1510	1807	2500	2812	3011	3902	4178	5902	5915
A	B	C	D	E	F	G	H	I	J	K	L	M
6102	6526	6818	7020	7155	7913	8016	8230	8599	8902	9090	9526	9812
N	O	P	Q	R	S	T	U	V	W	X	Y	Z

Your Ships

Number of Shots Used:

A	B	C	D	E	F	G	H	I	J	K	L	M
N	O	P	Q	R	S	T	U	V	W	X	Y	Z

Photocopiable for classroom use only.
© 2005 Computer Science Unplugged (www.unplugged.canterbury.ac.nz)

Photocopiable for classroom use only.
© 2002 Computer Science Unplugged (www.unplugged.canterbury.ac.nz)

My Ships

Number of Shots Used:

0	1	2	3	4	5	6	7	8	9
A 1982	C 6113		E 9121	H 5009	L 1248	O 2004	R 9369	V 3285	W 9172
B 7841	D 1055		F 1011	I 2651	M 1716	P 5173	S 1321		X 2052
			G 2984	J 1751	N 2148	Q 2806	T 3004		Y 6012
				K 4848			U 7190		Z 7525

Your Ships

Number of Shots Used:

0	1	2	3	4	5	6	7	8	9
A	E	H	K	L		O	R	V	Y
B	F	I		M		P	S	W	Z
C	G	J		N		Q	T	X	
D							U		

3A′

© 2005 Computer Science Unplugged (www.unplugged.canterbury.ac.nz)
Photocopiable for classroom use only.

My Ships

Number of Shots Used:

0	1	2	3	4	5	6	7	8	9
A 8615	E 1361	H 7726	K 3000	L 1814		O 9656	R 6993	V 8208	Y 2917
B 7003	F 7644	I 9003		M 2002		P 4002	S 3121	W 9423	Z 4122
C 1991	G 5600	J 5557		N 8844		Q 1221	T 4300	X 4176	
D 6211							U 1907		

Your Ships

Number of Shots Used:

0	1	2	3	4	5	6	7	8	9
A	C		E	H	L	O	R	V	W
B	D		F	I	M	P	S		X
			G	J	N	Q	T		Y
				K			U		Z

3B′

What's it all about?

Computers store a lot of information, and they need to be able to sift through it quickly. One of the biggest search problems in the world is faced by Internet search engines, which must search billions of web pages in a fraction of a second. The data that a computer is asked to look up, such as a word, a bar code number or an author's name, is called a *search key*.

Computers can process information very quickly, and you might think that to find something they should just start at the beginning of their storage and keep looking until the desired information is found. This is what we did in the Linear Searching Game. But this method is very slow—even for computers. For example, suppose a supermarket has 10,000 different products on its shelves. When a bar code is scanned at a checkout, the computer must look through up to 10,000 numbers to find the product name and price. Even if it takes only one thousandth of a second to check each code, ten seconds would be needed to go through the whole list. Imagine how long it would take to check out the groceries for a family!

A better strategy is *binary search*. In this method, the numbers are sorted into order. Checking the middle item of the list will identify which half the search key is in. The process is repeated until the item is found. Returning to the supermarket example, the 10,000 items can now be searched with fourteen probes, which might take two hundredths of a second—hardly noticeable.

A third strategy for finding data is called *hashing*. Here the search key is manipulated to indicate exactly where to find the information. For example, if the search key is a telephone number, you could add up all the digits in the number and take the remainder when divided by 11. In this respect, a hash key is a little like the check digits discussed in Activity 4—a small piece of data whose value depends on the other data being processed. Usually the computer will find what it is looking for straight away. There is a small chance that several keys end up in the same location in which case the computer will need to search through them until it finds the one it is seeking.

Computer programmers usually use some version of the hashing strategy for searching, unless it is important to keep the data in order, or unless an occasional slow response is unacceptable.

© 2002 Computer Science Unplugged (www.unplugged.canterbury.ac.nz)