

6. foglalkozás

Hajócsata – *Keresési algoritmusok*

Összegzés

A számítógépektől gyakran várjuk, hogy információkat találjanak meg nagy adatgyűjteményekben. Ezt gyors és hatékony módon kell elvégezniük. Ez a foglalkozás három különböző keresési módszert mutat be: lineáris keresés, bináris keresés és hash-elés.

Előismeretek

- ✓ Matematika: 3. vagy felsőbb osztály. Számok felfedezése: nagyobb, kisebb, egyenlő
- ✓ Geometria: 3. vagy felsőbb osztály. Alak és tér felfedezése: koordináták

Képességek

- ✓ Logikus gondolkodás

Korcsoport

- ✓ 9 és több év

Eszközök

Minden gyereknek szüksége lesz:

- ✓ Másolat a hajócsata játékokról
 - 1A, 1B az 1. játékhoz
 - 2A, 2B a 2. játékhoz
 - 3A, 3B a 3. játékhoz
- ✓ Esetleg szükség lehet néhány kiegészítő játéklapra is: 1A', 1B', 2A', 2B', 3A', 3B'.

Csatahajók

Bevezető foglalkozás

1. Válasszunk ki körülbelül 15 gyereket és állítsuk fel őket szemben az osztállyal. Minden gyerek kapjon egy lapot, rajta egy számmal (véletlenszerű sorrendben). A maradék osztály előtt ezek a számok maradjanak rejtve.
2. Adjunk néhány másik gyereknek egy tárolót négy vagy öt cukorkával. A feladatuk az lesz, hogy megtaláljanak egy adott számot. „Fizetniük” kell, hogy megnézzenek egy adott kártyát. Ha megtalálják a megfelelő számot, mielőtt felhasználnák az összes cukorkájukat, megtarthatják a maradékot.
3. Ha úgy kívánjuk, ezt megismételhetjük.
4. Most keverjük meg a lapokat és osszuk ki ismét. Most a gyerekeknek növekvő sorrendbe kell rendeződniük. Ismét végeztessük el a keresést.

Ha a számok rendezettek, jó stratégia, ha egy „fizetést” használnak arra, hogy a középső gyerek felfedje a lapját és a gyerekek felét kizárják. Ha megismétlik ezt az eljárást, mindössze három édesség is elegendő az adott szám megtalálásához. A hatékonyságnövekedés teljesen nyilvánvaló.

Foglalkozás

A gyerekek ráérezhetnek, hogyan keres a számítógép, ha csatahajó játékot játszanak. A játék alatt stratégiát kell kidolgozniuk, hogy behatárolják a hajókat.

Csatahajók – Egy *lineáris* keresési játék

Olvassuk fel a következő utasításokat a gyerekeknek

1. Alkossatok párokat. A pár egyik tagja az 1A lapot kapja, a másik az 1B-t. Ne mutassátok meg a lapotokat a partnereteknek!
2. Mindegyikőtök karikázzon be egy csatahajót a játéklapok felső részén és mondjátok meg a számát a partnereteknek.
3. Most felváltva próbáljátok kitalálni, hol van a párotok hajója. (A hajó betűjelét kell mondanotok és a partneretek megmondja a számát annak a hajónak, amit a mondott betű jelölt.)
4. Hány lövés kellett, hogy megtaláld a partnered hajóját? Ez a te pontszámod ebben a játékban.

(Az 1A' és 1B' lapok extra lapok, amelyeket azok a gyerekek kaphatják, akik még akarnak játszani vagy akik véletlenül meglátják a párjuk lapját. A 2A', 2B' és 3A', 3B' lapok későbbi játékokra szolgálnak.)

Követő megbeszélés

1. Milyen pontszámokat értek el?
2. Mi lenne a lehető legalacsonyabb és legmagasabb pontszám? (1 és 26, feltéve, ha a gyerekek nem lőnek kétszer ugyanarra a hajóra. Ezt a módszert „lineáris keresésnek” hívják, mert azt feltételezi, hogy minden pozíción végigmegyünk.)

Csatahajók – Egy *bináris* keresési játék

Utasítások

A játék ezen verziójának utasításai ugyanazok, mint az előző játékban, de a hajókon szereplő számok most növekvő sorrendben vannak. Magyarázzuk el ezt a gyerekeknek, mielőtt hozzáfognának.

1. Alkossatok párokat. Egyikőtök a 2A lapot kapja, a másikatok a 2B-t. **Ne** mutassátok meg a lapotokat a partnereteknek!
2. Mindegyikőtök karikázzon be egy csatahajót a játéklapok felső sorában és mondjátok meg a számát a párotnak.
3. Most felváltva próbáljátok ki, hol található a párotnak hajója. (A hajó betűjelét kell mondanotok és a partneretek megmondja a számát annak a hajónak, amit a mondott betű jelölt.)
4. Hány lövésre volt szükség, hogy megtaláljátok a párotnak hajóját? Ez a ti pontszámotok erre a játékra.

Követő megbeszélés

1. Milyen pontszámokat értetek el?
2. Milyen stratégiát használtak azok, akik a legkisebb pontszámot kapták?
3. Melyik hajót érdemes először választani? (A középső hajó megfelel a sort és megmondja, hogy melyik félben kell lenni a kiválasztott hajónak.) Utána melyik helyet érdemes választani? (A legjobb stratégia megint az, ha a szakasz középső hajóját választjuk ki a kiválasztott hajót tartalmazó területről.)
4. Ha ezt a stratégiát alkalmazzuk, hány lövésre van szükségünk, hogy megtaláljunk egy hajót? (Legfeljebb ötre).

Ezt a módszert „bináris keresésnek” hívjuk, mert a problémát mindig két részre osztja.

Csatahajók – Keresési játék *hash felhasználásával*

Utasítások

1. Mindenki fogjon egy lapot, mint az előző játékokban és mondja el a párjának a kiválasztott hajó számát.
2. Ebben a játékban ki lehet találni, melyik oszlopban van a hajó (0 és 9 között). Csak össze kell adni a hajó számjegyeit. Az összeg utolsó számjegye annak az oszlopnak a száma, amiben a hajó található. Például a 2345 számú hajó helyének megtalálásához össze kell adni a $2+3+4+5$ számokat, amelyek 14-et adnak. Az összeg utolsó számjegye 4, tehát a hajó a 4. oszlopban található. Most már tudod az oszlopot, tehát csak ki kell találnod, az oszlopban melyik hajó a kiválasztott. Ezt a technikát „hash”-elésnek hívják, mivel a számjegyek „össze vannak rakva”.
3. Most játsszuk a játékot az új keresési stratégiával. Lehet több játékot is játszani ugyanazzal a lappal – csak más oszlopot kell kiválasztani.

(Megjegyzés: a többi játékkal ellentétben, a 3A' és 3B' lapokat párban kell használni, mikor a hajók mintázatának az oszlopokban meg kell felelniük egymásnak.)

Követő megbeszélés

1. Gyűjtsük be a pontokat és beszéljük meg őket, ahogy a korábbiakban.
2. Melyik hajókat lehet nagyon gyorsan megtalálni? (Azokat, amik egyedül vannak az oszlopokban). Melyik hajókat nehezebb megtalálni? (Azokat, amelyek több másik hajót is tartalmazó oszlopban vannak.)
3. A három közül melyik keresési módszer a leggyorsabb? Miért?

A három különböző módszer közül melyek az egyes módszerek előnyei? (A második stratégia gyorsabb, mint az első, az elsőhöz viszont a hajóknak nem kell sorba rendezettnek lenniük. A harmadik stratégia rendszerint gyorsabb, mint a másik kettő, de véletlenül megeshet, hogy nagyon lassú lesz. A legrosszabb esetben az összes hajó ugyanabba az oszlopba kerül, így ugyanolyan lassú lesz, mint a legelső stratégia.)

Kiegészítő foglalkozások

1. A gyerekek készítsenek saját játékokat a három típus felhasználásával. A második játékhoz növekvő sorrendbe kell tenniük a számokat. Kérdezzük meg, hogy tennék a hash-elős játékot nagyon nehézre. (A legnehezebb akkor lesz, ha az összes hajó ugyanabba az oszlopba kerül). Hogyan tehetnénk a legkönnyebbé ugyanezt? (Igyekeznünk kell ugyanannyi számú hajót tenni minden egyes oszlopba.)
2. Mi van, ha az elsüllyesztett hajó nincs ott, ahol lennie kéne? (A lineáris kereséses játékban 26 lövés kell ennek kiderítéséhez. A bináris keresésesben pedig öt lövés, hogy bebizonyítsuk ezt. A hash-elős játék esetében attól függ, hány hajó található abban az oszlopban.)
3. A bináris keresési stratégiát használva hány lövésre van szükség, ha száz hely lenne (kb. hat lövés), ha ezer (kb. kilenc) vagy millió (kb. tizenkilenc)? (Vegyük észre, hogy a lövések száma nagyon lassan növekszik a hajók számához képest. Egy pluszlövés szükséges, amikor a méret duplázódik, tehát arányos a hajók számának logaritmusával.)

My Ships

Number of Shots Used:

A	9058	B	7169	C	3214	D	5891	E	4917	F	2767	G	4715	H	674	I	8088	J	1790	K	8949	L	13	M	3014
N	8311	O	7621	P	3542	Q	9264	R	450	S	8562	T	4191	U	4932	V	9462	W	8423	X	5063	Y	6221	Z	2244

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	

1A

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															

1B

My Ships		Number of Shots Used:																									
163	445	622	1410	1704	2169	2680	2713	2734	3972	4208	4871	5031															
A	B	C	D	E	F	G	H	I	J	K	L	M															
5283	5704	6025	6801	7440	7542	7956	8094	8672	9137	9224	9508	9663															
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															



























2A

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																	

2B

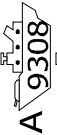


















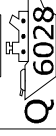






My Ships

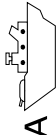
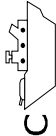

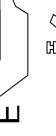
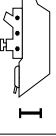
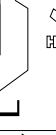
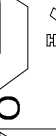
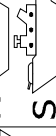

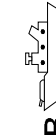
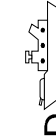

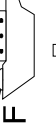

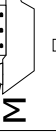
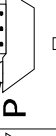
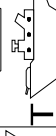

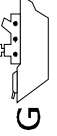
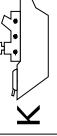
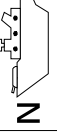
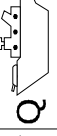

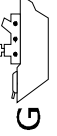
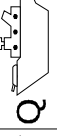

Number of Shots Used:									
0	1	2	3	4	5	6	7	8	9
A 	C 		E 	H 	L 	O 	R 	V 	W 
B 	D 		F 	I 	M 	P 	S 		X 
			G 	J 	N 	Q 	T 		Y 
				K 			U 		Z 

Your Ships

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

3A

My Ships		Number of Shots Used:								
0  9308  1478  8417  9434	1  6519  2469  5105	2  1524  8112  2000	3  4135	4  9050  1265  5711	5	6  4200  7153  6028	7  3121  9503  1114  7019	8  2385  5832  1917	9  1990  2502	

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

My Ships													Number of Shots Used:												
6123	1519	9024	5164	2038	2142	7156	9974	9375	7104	1004	1023	5108													
A	B	C	D	E	F	G	H	I	J	K	L	M													
1884	3541	5251	4840	3289	3654	2480	5602	8965	4053	2405	2304	1959													
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													

1A'

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															




















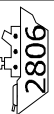




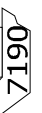

Your Ships		Number of Shots Used:																									
A	B	C	D	E	F	G	H	I	J	K	L	M															
N	O	P	Q	R	S	T	U	V	W	X	Y	Z															

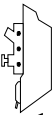
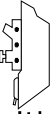
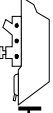
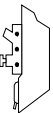
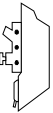


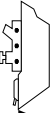
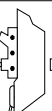




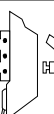
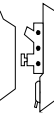







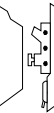

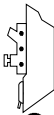
1B'

My Ships		Number of Shots Used:																									
28	326	943	1321	1896	2346	2430	2929	3106	3417	4128	4717	4915															
A	B	C	D	E	F	G	H	I	J	K	L	M															
5123	5615	6100	7015	7120	7695	7812	8103	8719	9020	9608	9713	9911															
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															

2A'

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 		W 9172 
B 7841 	D 1055 		F 1011 	I 2651 	M 1716 	P 5173 	S 1321 	V 3285 	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																			

3A'

Mire jó ez az egész?

A számítógépek rengeteg információt tárolnak és fontos, hogy képesek legyenek arra, hogy hamar előszedjék azokat. Az egyik legnagyobb keresési problémával a világon az internetes keresőmotorok szembesülnek, amelyeknek milliárdnyi weboldalt kell átnézniük egy szempillantás alatt. Azt az adatot, amit a számítógépnek adunk, hogy kikeresse, például egy szó, vonalkód vagy egy szerző neve, *keresési kulcsnak* nevezzük.

A számítógépek nagyon gyorsan fel tudják dolgozni az információkat és azt gondolhatnánk, hogy ahhoz, hogy megtaláljanak valamit, a tárolóeszközük elejétől kezdve végig kell nézni az egészet, amíg meg nem találják a kívánt információt. Ezt csináltuk a *lineáris keresés* játékbán. De ez a módszer nagyon lassú – még a számítógépeknek is. Például vegyünk egy szupermarketet, amely 10000 különböző terméket tart a polcain. Amikor a pénztárnál beolvassák egy vonalkódot, a számítógépnek végig kell néznie 10000 számot, hogy megtalálja a termék nevét és árát. Mégha csak egy ezredmásodpercig tart minden egyes kód ellenőrzése, tíz másodperc kell az egész lista átnézéséhez. Képzeljük el, mennyi ideig tartana, amíg egy család teljes bevásárlását végignézik!

A *bináris keresés* jobb stratégia. Ebben a módszerben a számokat rendezve tároljuk. A középsőt megnézzve, meg tudjuk állapítani, hogy melyik felében található a keresőkulcs. Ezt addig végezzük, amíg meg nem találjuk az adott elemet. Visszatérve a szupermarketes példára, a 10000 elemet most tizennégy próbával végig tudjuk nézni, ami összesen két századmásodpercbe telik – ezt észre sem venni.

Egy harmadik stratégiát az adat megtalálásához *hash-elésnek* hívják. Itt úgy állítják be a keresőkulcsot, hogy pontosan megadja azt a helyet, ahol az információ megtalálható. Ha tehát a keresési kulcs egy telefonszám, csinálhatjuk azt, hogy összeadjuk az összes számjegyet, elosztjuk 11-gyel és vesszük a maradékát. E tekintetben a hash-kulcs egy kicsit olyan, mint az ellenőrzőszámok a 4. foglalkozásban – egy kis adatdarab, amelynek értéke függ a feldolgozandó adattól. Általában a számítógép azonnal megtalálja, amit keres. Kicsi az esélye annak, hogy több kulcs is ugyanazt a helyet adja meg, amikor is a számítógépnek végig kell keresnie az egészet, hogy megtalálja az az egyetlen elemet, amit keres.

A számítógép-programozók általában valamiféle hash-elési stratégiát alkalmaznak a kereséshez vagy azért, mert fontos rendezetten tárolni az adatokat, vagy mert a lassú válaszidő elfogadhatatlan.