

De tweede programmeeropgave — N+1-Dames

Algoritmiëk voorjaar 2009, Universiteit Leiden

HET PROBLEEM

Het N-dames op schaakbord probleem luidt: gegeven een N bij N (schaak)bord, op hoeveel manieren kan men daar N dames op plaatsen zonder dat deze elkaar aanvallen? Zie de collegesheets van het vijfde college voor de probleemstelling, de spelregels en een backtracking-programma dat alle goede standen afdrukt.

Merk op dat als er een pion (of een ander stuk) tussen twee dames uit eenzelfde rij/kolom/diagonaal staat, deze twee dames elkaar niet meer kunnen slaan. Door toe te laten dat er pionnen tussen elkaar aanvallende dames staan, kunnen er meer dan N dames geplaatst worden zonder dat zij elkaar aanvallen. Dit leidt tot de volgende variant op het N-dames probleem, genaamd het **N+k-dames** probleem.

We hebben een N bij N bord met $N+k$ dames en k pionnen. Op hoeveel manieren kunnen de dames en de pionnen op het bord geplaatst worden zonder dat de dames elkaar kunnen slaan?

Om het iets gemakkelijker te maken zullen we ons in deze programmeeropdracht beperken tot het geval dat $k = 1$. We bekijken dus het **N+1-dames probleem**. Voor het geval dat $N = 8$ zijn er 128 goede standen. Een voorbeeld van zo'n goede stand met 9 dames op het 8 bij 8 bord is:

						D	
			D				
D							
				D			
	D					P	D
					D		
		D					
						D	

D = dame
P = pion

EEN ANALYSE

Er moet bij deze opdracht een verslag gemaakt worden, waarin onder andere onderstaande vragen beantwoord worden. Geef overal een duidelijke toelichting.

1. Bewijs dat voor een goede stand waarin geen van de $N + 1$ dames een andere dame kan slaan, geldt:
 - Er staat altijd precies 1 dame op een veld links van de pion en 1 dame op een veld rechts van de pion (dus in dezelfde rij)
 - Er staat altijd precies 1 dame op een veld boven de pion en 1 dame op een veld onder de pion (dus in dezelfde kolom)
2. Toon aan dat er nooit een pion op de rand van het bord kan staan (dus in de eerste of laatste rij of kolom).
3. Laat zien dat er ook geen pion kan staan op de vier velden die diagonaal grenzen aan de hoekvelden, dus bijv. het veld in de tweede rij en tweede kolom.

4. Bewijs dat er voor het $4 + 1$ -probleem, dus met 5 dames en 1 pion op een 4 bij 4 bord, geen goede standen bestaan.
5. Bewijs dat er voor het $5 + 1$ -probleem, dus met 6 dames en 1 pion op een 5 bij 5 bord, geen goede standen bestaan.

HET PROGRAMMA

Er dient een C++-programma geschreven te worden dat voor een door de gebruiker in te voeren waarde van N het aantal goede standen geeft van $N + 1$ dames en 1 pion op een N bij N bord. Het programma moet gebruik maken van *backtracking* (recursief of niet-recursief geïmplementeerd). Bekijk als voorbeeldprogramma het eerder genoemde programma voor het N -dames probleem. Het te schrijven programma is een aanpassing daarvan. Behalve het aantal goede standen moet ook een voorbeeld van een goede stand worden afgedrukt op het scherm.

Opmerkingen

- Gebruik bij het neerzetten van de dames in je backtrackfunctie wat je in vraag 1. en 2. van de analyse bewezen hebt.
- Gebruik een klasse `dames`, waarin behalve de representatie van een stand ook memberfuncties zoals `drukaf`, `geenaanval` en, natuurlijk, de backtrackfunctie staan.
- Bij het inlezen van N mag je gewoon gebruik maken van `cin>>getal`. Kies N niet te groot, want er zijn veel goede standen; voor $N = 15$ al meer dan 20 miljoen.
- Het aantal goede standen en een voorbeeld van een goede stand moeten op het beeldscherm worden getoond.
- Boven elke functie moet een commentaarblokje komen met daarin een (zeer) korte beschrijving van wat de functie doet. Noem daarin ook de gebruikte parameters: geef hun betekenis en geef aan hoe ze eventueel veranderd worden door de functie. Geef bij memberfuncties ook aan wat deze met de membervariabelen van het object doen. Let ook op de layout (consequent inspringen) en op het overige commentaar bij de programmacode (alleen zinvol en kort commentaar).
- Liefhebbers kunnen ook het algemenere $N + k$ -dames probleem oplossen. Dit is echter wel moeilijker dan het probleem met $k = 1$.
- Maak een getypt **verslag**, bij voorkeur in \LaTeX , waarin de in “EEN ANALYSE” gevraagde stellingen en bewijzen, alsmede een tabel met daarin voor zo veel mogelijke waarden van N het gevonden aantal goede standen. Leg in het verslag bovendien duidelijk (maar niet al te lang) uit hoe de methode backtracking voor het vinden van de goede standen werkt. Het verslag moet uiteraard ook een korte introductie van het probleem bevatten.
- Eventuele extra informatie over de programmeeropdracht is te vinden op:
<http://www.liacs.nl/home/mwitsenb/Algoritmiek/>
 Voor meer informatie over het vak, laatste wijzigingen, etcetera, raadplege men:
<http://www.liacs.nl/home/graaf/ALGO/>

Uiterste inleverdatum: vrijdag 17 april 2009. Voor elke week te laat inleveren gaat er een punt van het cijfer af. Het programma per e-mail sturen aan de hoofdnakijker: mwitsenb@liacs.nl. Listing en verslag moeten op papier worden ingeleverd en in de daartoe bestemde doos met opschrift Algoritmiek in de postkamer van Informatica (kamer 156) worden gedeponneerd. Vermeld overal duidelijk de namen van de makers.

Normering: verslag 3; commentaar en layout 1; modulaire opbouw en OOP 1; werking 5.