

TEST PRACTIC

(Problema Razboi Naval)

Autor: Mircea Digulescu

Enunt

Intr-un tinut indepartat doua mari puteri (Albastru si Rosu) se afla in mijlocul unei confruntari navale. Golful in care se desfasoara confruntarea este un dreptunghi de dimensiune $N \times N$ in care sunt situate Nave. Navele ambelor parti au ajuns in teatru de operatiuni inainte de declansarea confruntarii si deja ocupa anumite pozitii. Datorita vitezei cu care se desfasoara razboiul naval in zilele noastre, navele NU apuca sa se miste (sa isi schimbe pozitia) pe durata confruntarii. Toate Navele ocupa pozitii disjuncte (nu se intersecteaza). Ambele parti dispun de 4 categorii de Nave: Salupa, Submarin, Distrugator si Crucisator. Toate Navele au o Pozitie si o Forma (sunt alcatuite din mai multe patrute 1×1). Ele sunt echipate cu rachete de atac, avand eventual si alte abilitati speciale si sufera in mod diferit consecintele unui atac. Toate Navele dispun de un anumit nivel de Combustibil (intreg) si, ca urmare a operatiunilor de lupta, au suferit un anumit grad de avarie (0-100). O Nava care a ajuns la **Gradul de avarie 100** sau care a ajuns cu **Combustibilul la 0** se considera distrusa si nu mai participa la operatiuni.

Conflictul naval se desfasoara pe Randuri. In cadrul unui Rand, toate navele aleg (simultan) o actiune dintre cele pe care pot sa le intreprinda. Actiunea isi produce efectele la inceputul randului urmator. La finalul fiecarui Rand, o Nava consuma o cantitate de combustibil (in L) egala cu numarul de patrutele al Formei sale (e.g. pentru Distrugator, 3 Litri) pentru intretinere. Actiunea de *Regenerare* consuma **5 Litrii** de combustibil si scade *Gradul de Avarie* cu **14** (dar nu la mai putin de 0). Lansarea unei rachete de atac reduce stocul de astfel de rachete cu **1**. Se poate lansa maxim 1 racheta de atac / Rand de catre o Nava.

Mai multe informatii despre fiecare categorie de Nava se regasesc in tabelul de mai jos:

Informatii / Nava	Salupa	Submarin	Distrugator	Crucisator
Forma	*	* *	* * *	<div style="display: flex; justify-content: space-between;"> <div> ** ** ** </div> <div> ** sau ** </div> <div> ** **** ** </div> </div> (simplu) (cu sistem aparare)
Resurse	Combustibil Rachete de atac	Combustibil Rachete de atac	Combustibil Rachete de atac	Combustibil Rachete de atac
Grad de avarie in cazul atacului cu rachete	+100	+50	+35	+20* * in caz ca nu sunt interceptate (vezi mai jos)
Actiuni posibile	Nimic Atac cu rachete	Nimic Atac cu rachete Autodistrugere	Nimic Atac cu rachete Regenerare	Nimic* Atac cu rachete* Regenerare*

Nivel initial de combustibil	55	225	300	N/A (diferă de la nava la nava)
-------------------------------------	----	-----	-----	---------------------------------

* - Suplimentar, pe lângă acțiunea principală, Crucisatoarele dotate cu apărare antirachetă, pot opta să armeze acest sistem.

Precizări suplimentare:

- O acțiune a unei Nave este *disponibilă* (poate fi efectuată) doar dacă există suficiente Resurse pentru efectuarea ei. Utilizatorul va alege *doar dintre acțiunile disponibile!*
- O Nava cu grad de avarie de 70 sau mai mare, *nu poate lansa atacuri* acel Rand.
- Un Crucisator echipat cu sistem de apărare poate, la fiecare rand, să opteze suplimentar să Armeze acest sistem de apărare, pe lângă acțiunea obișnuită pe care o efectuează. Armarea sistemului consumă **2 Litri** de combustibil. Dacă acest sistem este armat, toate rachetele lansate în respectivul rand care ar lovi acel Crucisator, au o șansă de 75% să fie distruse, caz în care nu produc daune.
- Activarea mecanismului de Autodistrugere a unui Submarin conduce la distrugerea completă (avarie +100) a tuturor navelor ale căror forme se intersectează cu Zona Tintă. Zona țintă este alcătuită din toate patratelele ale căror coordonate (X și Y) diferă prin cel mult 2 de cele ale unui patratel din forma Submarinului.
- Regenerarea consumă **5 Litri de combustibil** și are drept efect reducerea gradului de avarie a respectivei Nave cu **14** (dar el nu poate scădea sub 0).
- Lansarea unei rachete de către o Nava presupune alegerea (de către Utilizator, de la tastatură) a patratelului (coordonate X și Y) țintă. Dacă Forma unei Nave cuprinde acel patratel, respectivă Nava se consideră țintită și suferă (la randul următor) consecințele specifice.

Lupta constă într-o succesiune de Randuri, cât timp există Nave nedistruse aparținând ambelor părți. Dacă la începutul unui Rand există Nave nedistruse aparținând unei singure părți, acea parte se consideră INVINGĂTOARE. Dacă la finalul unui Rand nu rămâne nici o Nava nedistrusă, lupta se consideră REMIZĂ.

Desfășurarea unui Rand:

În cadrul unui rand, următoarele au loc, în ordine:

1. Se afișează pe ecran numărul de Nave aparținând fiecărei părți care încă nu sunt distruse.
2. Regenerarea produce efecte asupra Navelor care au ales această acțiune Randul anterior.
3. Atacurile își produc efectele, eventual ducând la distrugerea unor Nave. De asemenea, Navele ramase fără combustibil sunt distruse în acest pas.
4. Fiecare Nava (a fiecărei părți), **încă nedistrusă**, optează pentru FIX o acțiune în randul curent (mai puțin Crucisator echipat cu sistem de apărare care poate, suplimentar opta să activeze și acest sistem). Acțiunea dorită este selectată de utilizator de la tastatură, *dintre cele disponibile*. Odată aleasă o acțiune (pentru o Nava), se introduc și eventualii parametrii suplimentari (e.g. coordonatele X,Y ale țintei atacului).
5. Navele consumă cantitatea de combustibil specifică pentru întreținere.

Cerinta

Realizați un program în C++ care, pornind de la o configurație inițială a câmpului de luptă (citită din fișier sau de la tastatură), simulează o luptă navală, conform specificațiilor din enunț, pe baza acțiunilor alese de utilizator de la tastatură pentru fiecare Nava la fiecare rand. La final afișați rezultatul luptei (Partea INVINGĂTOARE / REMIZĂ).

Aspectele pentru care nu exista o cerinta explicita (e.g. formatul datelor de intrare/interactiunii cu utilizatorul) ramane la latitudinea voastra. Formatul in care se citesc datele de intrare si structura interactiuni cu utilizatorul este la latitudinea voastra. Exemplul prezentat nu are caracter normativ in aceasta privinta.

In realizarea programului este obligatorie folosirea urmatoarelor concepte de OOP: **Clase, Mostenire (Derivare), Incapsulare, Alocare dinamica a memoriei (cu new), Functii virtuale.**

Suplimentar, folositi cel putin 2 din urmatoarele 5 concepte de OOP: Accesibilitatea (ascunderea) membrilor, Membrii static/const, Supraincercarea functiilor, Supraincercarea operatorilor, Conversii.

Atentie: Conceptele folosite trebuie sa fie aplicate in mod adecvat in rezolvarea cerintei problemei.

Evaluarea va tine cont de urmatoarele elemente ale programului realizat: Corectitudine, Calitatea Design-ului, Folosire adecvata si utila a Conceptelor de OOP, Calitatea Codului Sursa (Claritate, etc.).

Atentie: Sursele care nu se compileaza, vor fi notate cu **1p**, indiferent de continut. Este recomandat sa specificati in comentarii mediul de dezvoltare folosit.

Recomandari

In realizarea programului, comisia recomanda sa proiectati urmatoarele clase:

- **Forma** – clasa care reprezinta o colectie de blocuri ce alcatuiesc forma unei nave
- **Nava** – clasa *de baza* pentru Nave
- **Clase derivate pentru fiecare nava in parte**
- **Actiune** – clasa ce reprezinta o actiune aleasa de utilizator pentru o anumita Nava

Avand in vedere ca testul vizeaza evaluarea cunostintelor de programare OOP, este *suficient* sa determinati ce nava(e) se afla la o coordonata parcurgand de fiecare data toate Navele din teatrul de operatiuni, fara a tine vreo structura de date speciala pentru imbunatatirea performantei.

Precizari finale

- Timpul efectiv de lucru, inclusiv citirea enuntului este de **90 de minute**.
- Accesul la Internet este **interzis**.

Exemplu

Exemplu de date de intrare pentru configuratia initiala (1 submarin, 1 distrugator, 2 crucisatorare):

Rosu X 0 Y 1 Submarin 10 racheteatac | Albastru X 1 Y 2 Crucisator false 500 Litri 30 racheteatac | Rosu Crucisator X 1 Y 6 true 400 Litri 20 racheteatac | Albastru Distrugator X 10 Y 6 10 racheteatac .

Exemplu de interactiune cu Utilizatorul:

Randul 1: Computer: Albastru – 2 Nave, Rosu – 2 Nave; **Computer: Alegeti actiune pentru Submarin Rosu (X=0,Y=0) – Atac, Autodistrugere, Nimic; Utilizator: Autodistrugere;** Computer: Alegeti actiune pentru Crucisator Albastru (X=1,Y=2) – Atac, Regenerare, Nimic; Utilizator: Atac; Computer: Coordonatele tinta (X, Y)? Utilizator: 1 5; **Computer: Alegeti actiune pentru Crucisator Rosu (X=1,Y=6) – Atac, Regenerare, Nimic; Utilizator: Atac; Computer: Coordonatele tinta (X, Y)? Utilizator: 10 6; Computer: Doriti sa activati sistemul de aparare al Crucisatorului? Utilizator: Da.** Computer: Alegeti actiune pentru Distrugator Albastru (X=10,Y=6) – Atac, Regenerare, Nimic; Utilizator: Nimic.

Randul 2: Computer: Albastru – 1 Nave, Rosu – 1 Nave; **Computer: Alegeti actiune pentru Crucisator Rosu (X=1,Y=6) – Atac, Regenerare, Nimic; Utilizator: Atac; Computer: Coordonatele tinta (X, Y)? Utilizator: 10 6; Computer: Doriti sa activati sistemul de aparare al Crucisatorului? Utilizator: Nu.** Computer: Alegeti actiune pentru Distrugator Albastru (X=10,Y=6) – Atac, Regenerare, Nimic; Utilizator: Atac. Computer: Coordonatele tinta (X, Y)? Utilizator: 1 6.

Randul 3: Computer: Albastru – 1 Nave, Rosu – 1 Nave; **Computer: Alegeti actiune pentru Crucisator Rosu (X=1,Y=6) – Atac, Regenerare, Nimic; Utilizator: Atac; Computer: Coordonatele tinta (X, Y)? Utilizator: 10 6; Computer: Doriti sa activati sistemul de aparare al Crucisatorului? Utilizator: Da.** Computer: Alegeti actiune pentru Distrugator Albastru (X=10,Y=6) – Regenerare, Nimic; Utilizator: Nimic.

Randul 4: Computer: Albastru – 0 Nave, Rosu – 1 Nava. **ROSU ESTE INVINGATOR.**