**Facultatea de Matematica – Informatica**

**Programare Orientata Obiect (Anul I) 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 x 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 patrate 1x1). 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 patratele 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 \* \***

\*

\*

\*

\*

\*\* \*\*

\*\* sau \*\*\*\*

\*\* \*\*

(simplu) (cu sistem aparare)

Resurse - Combustibil

- Rachete de

atac

- Combustibil

- Combustibil

- Combustibil

- Rachete de atac

- Rachete de atac

- 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

- Nimic

- Atac cu rachete

- Atac cu rachete

- Autodistrugere

- Regenerare

- Nimic\*

- Atac cu rachete\*

- Regenerare\*

- 1 -

**Nivel initial de combustibil**

55 225 300 N/A (difera de la nava la

nava)

*\* - Suplimentar, pe langa actiunea principala, Crucisatoarele dotate cu aparare antiracheta, pot opta sa armeze acest sistem.*

Precizari suplimentare:

• O actiune a unei Nave este disponibila (poate fi efectuata) doar daca exista suficiente Resurse pentru efectuarea ei. Utilizatorul va alege doar dintre actiunile disponibile!

• O Nava cu grad de avarie de 70 sau mai mare, nu poate lansa atacuri acel Rand.

• Un Crucisator echipat cu sistem de aparare poate, la fiecare rand, sa opteze suplimentar sa Armeze acest sistem de aparare, pe langa actiunea obisnuita pe care o efectueaza. Armarea sistemului consuma 2 Litrii de combustibil. Daca acest sistem este armat, toate rachetele lansate in respectivul rand care ar lovi acel Crucisator, au o sansa de 75% sa fie distruse, caz in care nu produc daune.

• Activarea mecanismului de Autodistrugere a unui Submarin conduce la distrugerea completa (avarie +100) a tuturor navelor ale caror forme se intersecteaza cu Zona Tinta. Zona tina este alcatuita din toate patratelele ale caror coordonate (X si Y) difera prin cel mult 2 de cele ale unui patratel din forma Submarinului.

• Regenerarea consuma 5 Litri de combustibil si are drept efect reducerea gradului de avarie a respectivei Nave cu 14 (dar el nu poate scade sub 0).

• Lansarea unei rachete de catre o Nava presupune alegerea (de catre Utilizator, de la tastatura) a patratelului (coordonate X si Y) tinta. Daca Forma unei Nave cuprinde acel patratel, respectiva Nava se considera tintita si sufera (la randul urmator) consecintele specifice.

Lupta consta intr-o succesiune de Randuri, cat timp exista Nave nedistruse apartinand ambelor parti. Daca la inceputul unui Rand exista Nave nedistruse apartinand unei singure parti, acea parte se considera INVINGATOARE. Daca la finalul unui Rand nu ramane nici o Nava nedistrusa, lupta se considera REMIZA.

Desfasurarea unui Rand:

In cadrul unui rand, urmatoarele au loc, in ordine:

1. Se afiseaza pe ecran numarul de Nave apartinand fiecarei parti care inca nu sunt distruse.

2. Regenerarea produce efecte asupra Navelor care au ales aceasta actiune Randul anterior.

3. Atacurile isi produc efectele, eventual ducand la distrugerea unor Nave. De asemenea, Navele

ramase fara combustibil sunt distruse in acest pas.

4. Fiecare Nava (a fiecarei parti), inca nedistrusa, opteaza pentru FIX o actiune in randul curent (mai putin Crucisator echipat cu sistem de aparare care poate, suplimentar opta sa activeze si acest sistem). Actiunea dorita este selectata de utilizator de la tastatura, dintre cele disponibile. Odata aleasa o actiune (pentru o Nava), se introduc si eventualii parametrii suplimentari (e.g. coordonatele X,Y ale tintei atacului).

5. Navele consuma cantitatea de combustibil specifica pentru intretinere.

Cerinta

Realizati un program in C++ care, pornind de la o configuratie initiala a campului de lupta (citita din fisier sau de la tastatura), simuleaza o lupta navala, conform specificatiilor din enunt, pe baza actiunilor alese de utilizator de la tastatura pentru fiecare Nava la fiecare rand. La final afisati rezultatul luptei (Partea INVINGATOARE / REMIZA).

- 2 -

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, Supraincarcarea functiilor, Supraincarcarea 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 rachateatac | 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 Crucistatorului? 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 Crucistatorului? 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 Crucistatorului? 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.

- 3 -