

Tema 2

Cerințe globale și obligatorii

- Alocare dinamică a memorie;
- Separarea declarației unei clase de implementare acesteia;
- Crearea de constructori (folosind lista de inițializare) și destructori;
- Utilizare supraincărcării operatorilor, unde este cazul;
- Indentare și comentarea adecvată a codului;
- Utilizarea unei convenții de denumire a variabilelor, metodelor și claselor, cu specificarea acesteia;
- Utilizarea variabilelor și metodelor *const*, unde este cazul;
- Utilizarea claselor abstracte și metodelor virtuale în rezolvarea cerinței;
- Tema trebuie să compileze fără a utiliza anumite flag-uri de compilare (cu excepția cazurilor în care pentru compilare este necesară o anumită versiune de C++) și să respecte standardele C++ pentru sintaxă;
- Pentru punctaj maxim nu trebuie să se folosească verificări de tipuri. Pentru a gestiona mai multe tipuri de date în același bloc de instrucțiuni se vor folosi clase abstracte și metode virtuale;
- Strategia folosită pentru mutarea agenților trebuie să fie deterministă;
- Tipurile de agenți trebuie să difere cel puțin prin funcționalitățile oferite de suprascrierea metodelor virtuale;
- Deadline: 25 aprilie 23:59;

1 Survival Game

Avem o hartă pe care sunt distribuiți o serie de agenți. Fiecare agent își poate modifica poziția pe hartă. Dacă poziția nou aleasă este deja ocupată, atunci va exista o confruntare și unul dintre agenți va fi eliminat. Fiecare poziție neocupată pe hartă poate conține un item care conferă avantaje agentului care îl găsește. Aceste acțiuni pot fi efectuate până când pe hartă rămâne un singur agent. Cerințe:

- Creare a minim 3 tipuri de agenți;

- Creare de itemi care să ofere agenților diverse capabilități;
- Crearea unei hărți de minim 15 x 15;
- Distribuirea a cel puțin 7 agenți de fiecare tip;
- Creare unei clase care să gestioneze harta și desfășurarea unei runde;
- Creare unei modalități de a alege simularea succesivă a rundelor sau oprirea simulării la stadiul curent, cu afișarea configurației hărții la sfârșitul unei runde simulate;
- Afișarea detaliilor despre ce modificări au avut loc asupra hărții;

2 The Walk

Avem o hartă pe care se afla un robot (în colțul din stânga sus). Pe hartă exista o locație în care robotul trebuie să ajungă. La fiecare pas robotul alege o nouă poziție pe care să se mute, astfel încât noua poziție să îl aducă cu un pas mai aproape de locație finală. De asemenea pe hartă sunt distribuite capcane și itemi care pot ajuta robotul în călătoria lui. Unele item-uri pot fi folosite doar dacă tipul robotului este compatibil cu item-ul. Robotul poate executa o mutare de cel mult o poziție per rundă, iar noua poziție nu poate fi la o distanță mai mare de 1 față de poziția în care se afla. Însă robotul poate ”vedea” la maxim 4 poziții față de poziția lui curentă. Cerințe:

- Crearea a minim trei tipuri de roboti;
- Creare de itemi care să ofere diverse capabilități;
- Crearea unei hărți de minim 15 x 15;
- Creare unei clase care să gestioneze harta și desfășurarea unei runde;
- Creare unei modalități de a alege simularea succesivă a rundelor sau oprirea simulării la stadiul curent, cu afișarea configurației hărții la sfârșitul unei runde simulate;
- Afișarea detaliilor despre ce modificări au avut loc asupra hărții;

3 Treasure hunt

Avem o hartă pe care sunt plasate 3 comori pe poziții aleatorii și 4 căutători de comori în cele 4 colțuri. Fiecare căutător poate înainta o casuță pe rundă și nu poate trece peste casuțele deja explorate de el sau de ceilalți căutători. La terminarea fiecărei runde, despre fiecare căutător se va ști dacă a găsit sau nu o comoară. Primul căutător care a găsit comoara va fi câștigător și va părăsi harta. Jocul continuă până când nu mai sunt casuțe de explorat sau nu mai exista mutări valide (toate casuțele explorate) sau rămâne doar un căutător pe hartă fără nici o comoară. Cerințe:

- Crearea a 4 tipuri de căutători de comori poziționați în colțurile hărții;

- Crearea a 3 comori așezate aleatoriu pe harta, dar niciodată în unul din cele 4 colțuri;
- Crearea unei hărți de cel puțin 15 x 15;
- Crearea unei clase care să gestioneze harta și desfășurarea unei runde;
- Crearea unei modalități de a alege simularea succesivă a rundelor sau oprirea simulării la stadiul curent, cu afișarea configurației hărții la sfârșitul unei runde simulate;
- Afișarea detaliilor despre ce modificări au avut loc asupra hărții;

4 The Shootout

Avem o harta pe care sunt plasați o serie de agenți. Fiecare agent are o arie de vizibilitate în care poate observa alți agenți, o arma care poate trage într-un anumit mod, în funcție de tipul ei, și (posibil) o armură care poate proteja agentul și care poate influența felul în care agentul folosește arma. Dacă agentul nu are nici un alt agent în aria de vizibilitate, atunci acesta își poate schimba poziția (mutarea se face cu o distanță mai mică decât aria de vizibilitate și direct proporțională cu aceasta). Toată acțiunea se desfășoară pe runde. Pe durata unei runde, fiecare agent poate folosi arma sau alege altă poziție, în funcție de contextul în care se află. Cerințe:

- Crearea a minim un tip de agent;
- Crearea a minim 3 tipuri de arme și crearea de armuri armuri;
- Crearea unei hărți de cel puțin 25 x 25;
- Crearea unei clase care să gestioneze harta;
- Crearea unei modalități de a alege simularea succesivă a rundelor sau oprirea simulării la stadiul curent, cu afișarea configurației hărții la sfârșitul unei runde simulate;
- Afișarea detaliilor despre ce modificări au avut loc asupra hărții;